

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

ФАКУЛЬТЕТ ЕЛЕКТРОНІКИ

(повна назва інституту/факультету)

КОНСТРУЮВАННЯ ЕЛЕКТРОННО-ОБЧИСЛЮВАЛЬНОЇ АПАРАТУРИ

(повна назва кафедри)

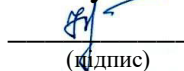
«На правах рукопису»

УДК: 003.26;

004.056.55

«До захисту допущено»

Завідувач кафедри

 О.М.Лисенко
(підпис) (ініціали, прізвище)

“_18_” грудня 2020 р.

Магістерська дисертація

зі спеціальності (спеціалізації) 172 – Телекомунікації та радіотехніка
(код і назва)

на тему: Методи підвищеної ефективності розпізнавання об'єктів на інфрачервоних зображеннях та відеопослідовностях в оптико-електронних системах

Виконав: студент II курсу, групи ДК-91мп
(шифр групи)

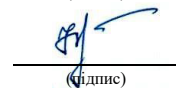
Лисенко Андрій Сергійович

(прізвище, ім'я, по батькові)


(підпис)

Науковий керівник: зав. каф., д.т.н., проф. Лисенко О.М.

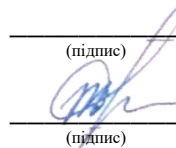
(посада, науковий ступінь, вчене звання, прізвище та ініціали)


(підпис)


Консультант _____
(назва розділу) (науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

Рецензент: доцент каф. АМЕС, к.т.н., Попович П.В.
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали)


(підпис)

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.

Студент 
(підпис)

Київ – 2020 року

**Національний технічний університет України
“Київський політехнічний інститут
імені Ігоря Сікорського”**

Інститут/факультет електроніки

(повна назва)

Кафедра конструювання електронно-обчислювальної апаратури

(повна назва)

Рівень вищої освіти – другий (магістерський) за освітньо - професійною програмою

Спеціальність 172 – Телекомунікації та радіотехніка

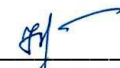
(шифр і назва)

ЗАТВЕРДЖУЮ
В.о. директора НДІ ЕМСТ
КПІ ім. Ігоря Сікорського

(підпис) Татарчук Д.Д.
(прізвище та ініціали)
«25» жовтня 2019 року

ЗАТВЕРДЖУЮ

Завідувач кафедри


(підпис) Лисенко О.М.
(прізвище та ініціали)

«23» жовтня 2019 року

ЗАВДАННЯ

**на магістерську дисертацію
студенту Лисенку Андрію Сергійовичу**

(прізвище, ім'я, по батькові)

1. Тема дисертації Методи підвищеної ефективності розпізнавання об'єктів на інфрачервоних зображеннях та відеопослідовностях в оптико-електронних системах

науковий керівник дисертації: Лисенко Олександр Миколайович, д.т.н., проф.,
(Прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від «05» листопада 2020 року № 3241-с

2. Термін подання студентом дисертації: 17 грудня 2020 року

3. Об'єкт дослідження: процеси цифрової обробки зображень із використанням згорткових нейронних мереж при розпізнаванні об'єктів

4. Предмет дослідження: методи розпізнавання об'єктів на інфрачервоних зображеннях та відеопослідовностях і алгоритмічні рішення їх реалізації.

5. Перелік завдань, які потрібно розробити: 1. Аналіз факторів, які впливають на виявлення об'єктів в інфрачервоному і видимому діапазонах. 2. Аналіз відомих технічних рішень побудови об'єкта дослідження за результатами проведеного патентного пошуку. 3. Порівняльний аналіз сучасних методів згорткових нейронних мереж розпізнавання об'єктів в режимі реального часу 4. Дослідження

різновидів архітектур YOLO та обґрунтування вибору моделі нейронної мережі 5. Розробка алгоритмів підвищення ефективності навчання моделей YOLO на персональному наборі даних 6. Моделювання та оцінка результатів навчання моделі нейронної мережі підвищеної ефективності 7. Розроблення стартап-проєкту

6. Орієнтовний перелік графічного (ілюстративного) матеріалу:
Презентація у форматі PowerPoint.

7. Орієнтовний перелік публікацій: 1 публікація.

8. Консультанти розділів дисертації


Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

9. Дата видачі завдання: 23 жовтня 2019р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1	Аналіз факторів, які впливають на виявлення об'єктів в інфрачервоному і видимому діапазонах	23.10.19-17.12.19	Вик.
2	Аналіз відомих технічних рішень побудови об'єкта дослідження за результатами проведеного патентного пошуку	18.12.19-01.02.20	Вик.
3	Порівняльний аналіз сучасних методів згорткових нейронних мереж розпізнавання об'єктів в режимі реального часу	02.02.20-15.05.20	Вик.
4	Дослідження різновидів архітектур YOLO та обґрунтування вибору моделі нейронної мережі	16.05.20-05.08.20	Вик.
5	Розробка алгоритмів підвищення ефективності навчання моделей YOLO на персональному наборі даних	06.08.20-19.10.20	Вик.
6	Моделювання та оцінка результатів навчання моделі нейронної мережі підвищеної ефективності	20.10.20-24.11.20	Вик.
7	Розроблення стартап-проєкту	25.11.20-30.11.20	Вик.
8	Оформлення дисертації	01.12.20-16.12.20	Вик.


Студент


(підпис)

Лисенко А.С.

(прізвище, ініціали)

Науковий керівник дисертації


(підпис)

Лисенко О.М.

(прізвище, ініціали)

РЕФЕРАТ

Магістерська дисертація складається з 99 сторінок, в якій міститься 40 рисунків, 27 таблиць, використано 27 джерел.

Актуальність. На сьогодні сучасні методи виявлення об'єктів на основі згорткових нейронних мереж забезпечують високу точність і продуктивність розпізнавання на високоякісних RGB зображеннях. Однак, застосування їх до даних низької якості та в іншому спектральному діапазоні призводить до суттєвого зниження ефективності, під якою розуміється швидкість обробки зображень у реальному часі (вище 30 кадрів/с), особливо на вбудованих пристроях (наприклад, в автономних транспортних засобах, інтерфейсах доповненої реальності, оптико-електронних системах відеоспостереження й безпеки тощо) та значення точності виявлення класів mAP вище за 85%. Ще одна складність, із якою можна зіткнутися під час навчання нейронної мережі – низький обсяг даних у порівнянні з поширеними зображеннями в RGB домені. Сукупність цих факторів визначає *актуальність* вирішуваного завдання.

Зв'язок роботи з науковими програмами, планами, темами. Дисертаційне дослідження проводилося в рамках д/б НДР №2116-п «Багатоканальний тепловізійно-телевізійний комплекс пошуку-виявлення із завадостійким швидкісним інтерфейсом передачі даних» відповідно пріоритетному напрямку розвитку науки і техніки України «Інформаційні та комунікаційні технології».

Метою дисертаційної роботи є підвищення ефективності (за критеріями точності та швидкодії) розпізнавання об'єктів на інфрачервоних зображеннях та відеопослідовностях шляхом подальшого розвитку методу розпізнавання об'єктів на основі згорткових нейромереж типу YOLO та алгоритмічних і програмних рішень його реалізації.

Досягнення мети вирішувалося наступними **задачами**:

- проаналізовано проблеми, які виникають при виявленні об'єктів в оптико-електронних тепловізійних пристроях; досліджено відомі запатентовані технічні рішення, які здійснюють виявлення об'єктів в інфрачервоному діапазоні; наведено огляд сучасних методів виявлення об'єктів при роботі в реальному часі;
- досліджено архітектури та проведено порівняльний аналіз двох передових моделей YOLO з метою обґрунтування вибору базового для використання;
- удосконалено обраний метод виявлення та проведено оцінку отриманих результатів.
- виконано розроблення стартап-проєкту.

Об'єктом дослідження є процеси цифрової обробки зображень з використанням глибоких згортальних нейронних мереж для розпізнавання об'єктів.

Предметом дослідження є методи розпізнавання об'єктів на інфрачервоних зображеннях та відеопослідовностях і алгоритмічні рішення їх реалізації.

Методи дослідження: методи виявлення об'єктів SSD, FPN, RetinaNet, YOLO; методи аугментації зображень CutMix, Mosaic; блоки та модулі для запобігання від перенавчання глибоких нейронних мереж DropBlock, Label Smoothing, Self-Adversarial Training; функції втрат та активації DIoU, GIoU, CIoU, Mish, Swish; архітектури об'єднання шарів глибоких нейронних мереж CSP, SAM, SPP; пакетна нормалізація CmBN, косінусна норма навчання, еволюційні алгоритми вибору гіперпараметрів.

Наукова новизна отриманих результатів визначається удосконаленням методом розпізнавання об'єктів на основі нейромережі YOLOv4 шляхом попередньої нормалізації і конвертації анотацій зображень навчальної вибірки, дублювання зображень навчальної вибірки з однаковим класом та навчання мережі з використанням попередньо навчених вагових коефіцієнтів згортки, що дозволило підвищити ефективність (за критеріями точності та швидкодії) розпізнавання об'єктів між класами на інфрачервоних зображеннях та відеопослідовностях.

Практичне значення отриманих результатів визначається створеними алгоритмічними та програмними засобами реалізації удосконаленого методу підвищення ефективності розпізнавання згорткових мереж на базі архітектури YOLO. Проведено також вивчення та порівняльне моделювання нейронних мереж із застосуванням запропонованих рішень.

Планується впровадження одержаних результатів в організації КП СПБ «Арсенал» (м. Київ), яке виконувалось в рамках та за завданням НДР КПІ ім. Ігоря Сікорського.

Апробація результатів дослідження. Результати дисертаційних досліджень доповідались і обговорювались в XIII Міжнародній науково-технічній конференції молодих вчених «Електроніка – 2020» (14-16 квітня 2020р.).

Публікації. За матеріалами досліджень опубліковано 1 друковану статтю:

- Лисенко А.С. Аналіз факторів, які впливають на ефективність оптико-електронних систем // Електронна та Акустична Інженерія. – 2020. – Т. 3, № 1. – С. 15 – 18 (див. Додаток А).

Ключові слова: теплове та інфрачервоне зображення; виявлення та розпізнавання об'єктів; згорткові нейронні мережі; YOLO.

ABSTRACT

The master's thesis consists of 99 pages, 40 figures, 27 tables, 27 refs.

Today, modern methods of object detection based on convolutional neural networks offer excellent recognition accuracy and performance on high quality RGB images. However, applying them to low-quality data with a different spectral range leads to a significant drop in efficiency. In this case, efficiency refers to the speed of real-time image processing (above 30 frames/s), especially on embedded devices (e.g., autonomous vehicles, augmented reality interfaces, optoelectronic video surveillance and security systems, etc.) and high value of mean average precision (mAP should be higher than 85%). Another difficulty you may encounter when training a neural network is the low amount of data compared to common images in the RGB domain. The combination of these factors forms the *relevant* problem.

Work relationship with scientific programs, plans, topics. Dissertation research was conducted within the d/b R&D №2116-p «Multi-channel thermal-television imaging complex search-detection with noise-resistant high-speed data transmission interface» on the priority direction of science and technology of Ukraine «Information and communication technologies».

The purpose of this thesis is to increase the efficiency (in terms of precision and speed) of object detection on infrared images and video sequences by further developing the method of object detection based on convolutional neural networks such as YOLO and algorithmic and software solutions for its implementation.

The following *tasks* were used to achieve the goal:

- Occurring problems were analyzed while detecting objects in opto-electronic thermal imaging devices; Prior patent solutions for infrared object detection were examined; State-of-the-art real-time object detection methods were reviewed.
- Two cutting-edge YOLO detectors architectures were researched as well as benchmarking study was conducted in order to make a specific choice.
- Object detection method was improved and the results were taken an assessment.
- Start-up project was developed.

The object of research is digital image processing using deep convolutional neural networks for object detection.

The subject of research is object detection methods for infrared images and video sequences and their implementation algorithms.

The methods of research are SSD, FPN, RetinaNet, YOLO object detection methods; CutMix and Mosaic image augmentation methods; DropBlock, Label Smoothing, Self-Adversarial Training blocks and modules for protection from overfitting deep neural networks; DIoU, GIoU, CIoU, Mish, Swish loss and activation functions; CSP, SAM, SPP deep neural network layer pooling architectures; CmBN batch normalization, cosine annealing scheduler, evolutionary hyperparametric selection algorithms.

The scientific novelty of the obtained results is determined by the improved method of object detection based on the YOLOv4 neural network via preliminary normalizing and converting annotations of training sample images, the same class training set images duplication and network training using pre-trained convolution weight coefficients, which improved the efficiency (in terms of precision and speed criteria) of object detection between classes on the infrared images and video sequences.

The practical importance of the obtained results is determined by the created algorithmic and software implementations improved method of increasing the efficiency of neural networks detection based on the YOLO architecture. The study and comparative benchmarking of neural networks using the proposed solutions were conducted.

The results obtained in the dissertation are planned to implement in the KP SPB «Arsenal» (Kyiv) organization, which was carried out within the assignment of the Igor Sikorsky's KPI R&D work.

The results of the thesis research were *reported and discussed* at the XIII International Scientific and Technical Conference of Young Scientists "Electronics - 2020" (April 14-16, 2020).

One *article* was published on the materials of the research:

- A. Lysenko. Analysis of Factors Affecting the Efficiency of Electro-Optical Systems// Електронна та Акустична Інженерія. – 2020. – Т. 3, № 1. – P.P.15 - 18.

Keywords: thermal and infrared imaging; object detection and recognition; convolutional neural networks; YOLO.

ЗМІСТ

Перелік умовних позначень, скорочень і термінів	3
ВСТУП.....	7
РОЗДІЛ 1. АНАЛІЗ ЛІТЕРАТУРНИХ ДЖЕРЕЛ ПО ТЕМІ ДОСЛІДЖЕННЯ	11
1.1. Фактори, які впливають на виявлення об'єктів в інфрачервоному і видимому діапазонах.	11
1.2 Аналіз відомих технічних рішень побудови об'єкта дослідження за результатами проведеного патентного пошуку	14
1.3 Порівняльний аналіз сучасних методів згорткових нейронних мереж розпізнавання об'єктів в режимі реального часу та постановка завдання дослідження	21
1.3.1 Single Shot Detector (SSD)	22
1.3.2 Feature Pyramid Network (FPN)	25
1.3.3 RetinaNet.....	26
1.3.4 You Only Look Once (YOLO)	28
Висновки за розд. 1	32
РОЗДІЛ 2. ДОСЛІДЖЕННЯ РІЗНОВИДІВ АРХІТЕКТУР YOLO ТА ОБҐРУНТУВАННЯ ВИБОРУ МОДЕЛІ НЕЙРОННОЇ МЕРЕЖІ	33
2.1 Архітектура і конфігурація YOLOv4	33
2.1.1 Базова частина	33
2.1.2 Шийна частина.....	36
2.1.3 Методи збільшення точності в базовій частині	38
2.1.4 Методи збільшення швидкості в базовій частині	40
2.1.5 Методи збільшення точності в головній частині	41

2.1.6 Методи збільшення швидкості в головній частині	45
2.2 Архітектура і конфігурація YOLOv5	45
2.3 Порівняльний аналіз версій YOLOv4 та YOLOv5	47
Висновки за розд. 2	55
РОЗДІЛ 3. МЕТОД ТА АЛГОРИТМИ ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ	
НАВЧАННЯ МОДЕЛЕЙ YOLO НА ПЕРСОНАЛЬНОМУ НАБОРІ	
ДАНИХ.	57
3.1 Оптимізація і конвертація анотацій зображень	57
3.2 Додаткова аугментація зображень.....	58
3.3 Роль рамок прив'язки в YOLO та їх налаштування	59
3.4 Налаштування конфігурації нейронної мережі	60
3.5 Моделювання та оцінка результатів навчання моделі нейронної мережі підвищеної ефективності.....	61
Висновки за розд. 3	65
РОЗДІЛ 4. РОЗРОБЛЕННЯ СТАРТАП-ПРОЄКТУ	
4.1. Сутність ідеї стартапу та технологічного рішення	66
4.2. Техніко-економічна характеристика продукції	67
4.3. Актуальність проєкту та аналіз ринку.....	68
4.4. Розроблення ринкової та маркетингової стратегії проєкту.....	74
Висновки за розд. 4	80
ЗАГАЛЬНІ ВИСНОВКИ.....	81
Список використаних джерел.....	83
Додаток А Копія публікації у науковому виданні	86
Додаток Б Лістинг програми	90

Перелік умовних позначень, скорочень і термінів

Аугментація даних

(augmentation) – методика створення додаткових навчальних даних із уже існуючих даних.

Базова частина

(backbone) – частина згорткової нейронної мережі для розпізнавання, яка об'єднує і формує характеристики зображення з різним ступенем структурованості.

Виділення ознак

(feature extraction) – процес зниження розмірності, в якому початковий набір початкових змінних скорочується до більш керованих груп (ознак) для подальшої обробки, водночас залишаючись достатнім набором для точного й повного опису початкового набору даних

Виключення

(dropout) – метод регуляризації штучних нейронних мереж, призначений для запобігання перенавчання мережі.

Гіперпараметри

(hyperparameters) – параметри, значення яких використовується для керування процесом навчання.

Головна частина

(head) – частина згорткової нейронної мережі для розпізнавання, що приймає дані із шийної частини і прогнозує класи об'єктів і їхні обмежувальні рамки.

Карта ознак

(feature map) – результат одного фільтра, застосовуваного до попереднього шару.

Ковзаюче вікно

(sliding window)

– в контексті комп'ютерного зору, прямокутна область фіксованої ширини і висоти, яка «ковзає» по зображенню для обчислення скалярного добутку матриці.

Максимізаційне

агрегування

(max pooling)

– процес дискретизації на основі вибірки, який передбачає обчислення максимального значення для кожного фрагмента карти ознак.

Метод зворотного

поширення помилки

(backpropagation)

– алгоритм навчання нейронної мережі прямого поширення.

Обмежувальні

рамки/прямокутники

(bounding boxes)

– уявні прямокутники, які оточують об'єкти, що необхідно розпізнати.

Ознаки (features)

– інформативні та незалежні властивості або характеристики спостережуваного явища, які застосовуються в алгоритмах розпізнавання образів і класифікації.

Пакетна

нормалізація (batch

normalization)

– метод, який використовується для прискорення й підвищення стабільності штучних нейронних мереж завдяки нормалізації вхідного шару через повторне центрування й масштабування.

Перенавчання

(overfitting)

– явище, коли навчена модель добре розпізнає приклади з навчальної множини, але не розпізнає або погано розпізнає будь-які інші приклади, які не брали участі в процесі навчання.

Придушення

немаксимумів

(Non-Maximum

Suppression)

– метод у розпізнаванні об'єктів, використовуваний для очищення декількох виявлень одного об'єкта.

Рамка/прямокутник

/поле прив'язки

(anchor box)

– набір передзаданих обмежувальних рамок певної висоти й ширини.

Регуляризація

(regularization)

– процес додавання деякої додаткової інформації, щоб уникнути перенавчання.

Рецептивне поле

(receptive field)

– частина матриці вхідних сигналів, що піддається операції згортки.

Усереднювальне

агрегування (average

pooling)

– процес дискретизації на основі вибірки, який передбачає обчислення середнього значення для кожного фрагмента карти ознак.

Шийна частина

(neck)

– послідовність шарів згорткової нейронної мережі для розпізнавання, в яких змішуються й комбінуються характеристики зображення, щоби передати їх для прогнозування.

ГНН	– глибока нейронна мережа
ЗНН	– згорткова нейронна мережа
ІЧ	– інфрачервоний
ОЕС	– оптико-електронні системи
FPS	– frames per second
GPU	– graphics processing unit
IoU	– Intersection over Union
LWIR	– Long Wavelength Infrared
mAP	– mean Average Precision
MWIR	– Medium Wavelength Infrared
NIR	– Near-infrared
RGB	– Red Green Blue
SWIR	– Short Wavelength Infrared

ВСТУП

Актуальність. Технологія формування теплових зображень генерує сцену на основі різниці інфрачервоного випромінювання між об'єктом і фоном. Тепловізійне зображення має низку переваг перед камерами та лідачами, оскільки вони отримують інформацію у видимому діапазоні спектра й гірше справляються з поганими погодними умовами та темними часами доби.

Тепловізійні датчики захоплення зображень, що фіксують навколишню ситуацію систем безпеки, транспортних засобів або безпілотних літальних апаратів мають безпомилково розуміти оточення. Однак, виявлення об'єктів на інфрачервоних зображеннях є більш складною проблемою у порівнянні зі звичайними камерами видимого діапазону. Хоча тепловізори мають переваги в темний час доби й у ситуаціях із поганими погодними умовами, вони страждають від низької якості роздільної здатності вихідного зображення.

На сьогодні сучасні методи виявлення об'єктів на основі згорткових нейронних мереж забезпечують високу точність і продуктивність розпізнавання на високоякісних RGB зображеннях. Однак, застосування їх до даних низької якості та в іншому спектральному діапазоні призводить до значного падіння ефективності. В даному випадку, під *ефективністю* мається на увазі швидкість обробки зображень у реальному часі (вище 30 кадрів/с), особливо на вбудованих пристроях (наприклад, в автономних транспортних засобах, інтерфейсах доповненої реальності, оптико-електронних системах відеоспостереження й безпеки тощо) та значення точності виявлення класів mAP вище за 85%.

Ще одна складність, із якою можна зіткнутися під час навчання нейронної мережі – низький обсяг даних у порівнянні з поширеними зображеннями в RGB домені. Сукупність цих факторів визначає *актуальність* вирішуваного завдання.

Зв'язок роботи з науковими програмами, планами, темами. Дисертаційне дослідження проводилося в рамках д/б НДР №2116-п «Багатоканальний тепловізійно-телевізійний комплекс пошуку-виявлення із завадостійким швидкісним інтерфейсом передачі даних» відповідно пріоритетному напрямку розвитку науки і техніки України «Інформаційні та комунікаційні технології».

Метою дисертаційної роботи є підвищення ефективності (за критеріями точності та швидкодії) розпізнавання об'єктів на інфрачервоних зображеннях та відеопослідовностях шляхом подальшого розвитку методу розпізнавання об'єктів на основі згорткових нейромереж типу YOLO та алгоритмічних і програмних рішень його реалізації.

Досягнення мети вирішувалося наступними **задачами**:

- аналіз проблем, які виникають при виявленні об'єктів в оптико-електронних тепловізійних пристроях; дослідження відомих патентних технічних рішень, які забезпечують виявлення об'єктів в інфрачервоному діапазоні; огляд сучасних методів виявлення об'єктів, що призначені для роботи в реальному часі.
- дослідження архітектури та порівняльний аналіз двох передових моделей YOLO з метою обґрунтування вибору базового для використання.
- удосконалення обраного методу виявлення та оцінка отриманих результатів.
- розроблення стартап-проєкту.

Об'єкт дослідження – процеси цифрової обробки зображень з використанням глибоких згортальних нейронних мереж для розпізнавання об'єктів.

Предмет дослідження – методи розпізнавання об'єктів на інфрачервоних зображеннях та відеопослідовностях і алгоритмічні рішення їх реалізації.

Методи дослідження – методи виявлення об’єктів SSD, FPN, RetinaNet, YOLO; методи аугментації зображень CutMix, Mosaic; блоки та модулі для запобігання від перенавчання глибоких нейронних мереж DropBlock, Label Smoothing, Self-Adversarial Training; функції втрат та активації DIoU, GIoU, CIoU, Mish, Swish; архітектури об’єднання шарів глибоких нейронних мереж CSP, SAM, SPP; пакетна нормалізація CmBN, косінусна норма навчання, еволюційні алгоритми вибору гіперпараметрів.

Наукова новизна отриманих результатів визначається удосконаленням методом розпізнавання об’єктів на основі нейромережі YOLOv4 шляхом попередньої нормалізації і конвертації анотацій зображень навчальної вибірки, дублювання зображень навчальної вибірки з однаковим класом та навчання мережі з використанням попередньо навчених вагових коефіцієнтів згортки, що дозволило підвищити ефективність (за критеріями точності та швидкодії) розпізнавання об’єктів між класами на інфрачервоних зображеннях та відеопослідовностях.

Практичне значення отриманих результатів визначається створеними алгоритмічними та програмними засобами реалізації удосконаленого методу підвищення ефективності розпізнавання згорткових мереж на базі архітектури YOLO. Проведено вивчення та порівняльне моделювання нейронних мереж із застосуванням запропонованих рішень.

Планується впровадження одержаних результатів в організації КП СІБ «Арсенал» (м. Київ), яке виконувалось в рамках та за завданням НДР КПІ ім. Ігоря Сікорського.

Апробація результатів дослідження. Результати дисертаційних досліджень доповідались і обговорювалися в XIII міжнародної науково-технічної конференції молодих вчених «Електроніка – 2020» (14-16 квітня 2020 р.).

Публікації. За матеріалами досліджень опубліковано 1 друковану статтю (див. Додаток А):

– Лисенко А.С. Аналіз факторів, які впливають на ефективність оптико-електронних систем // Електронна та Акустична Інженерія. – 2020. – Т. 3, №1. - С.15 – 18.

Структура дисертаційної роботи містить вступ, 4 розділи, загальний висновок, перелік використаної літератури та додатки:

- Додаток А. Копія публікації у науковому виданні;
- Додаток Б. Лістинг програми.

РОЗДІЛ 1. АНАЛІЗ ЛІТЕРАТУРНИХ ДЖЕРЕЛ ПО ТЕМІ ДОСЛІДЖЕННЯ

1.1. Фактори, які впливають на виявлення об'єктів в інфрачервоному і видимому діапазонах.

Оптико-електронні системи (ОЕС), які використовуються для пошуку та виявлення наземних об'єктів і повітряних носіїв стають усе більш поширеними з кожним роком. Водночас, сучасні алгоритми й методи комп'ютерного зору привнесли величезне значення для систем, які можуть обробляти зображення та відеодані так само, як люди або навіть краще. Якщо об'єднати два цих ствердження, то отримаємо високоефективні пошуково-оглядові комплекси, які можуть допомогти реєструвати відеодані в автономних транспортних засобах та системах для допомоги водію, а також у системах безпеки і спостереження. Однак, наразі в основному використовуються камери видимого діапазону спектра, які мають певні проблеми при виявленні об'єктів у темний час доби і при поганих погодних умовах. І разом із цим, камери, які працюють в інфрачервоному діапазоні можуть істотно підвищити показник виявлення у порівнянні з камерами видимого діапазону в подібних умовах. Це досягається завдяки тому, що тепловізори сприймають випромінювання з довжинами хвиль у діапазоні від 1,4 мкм до 14 мкм, які не сприймає людина. Відповідно, ці зображення відрізняються від того, що ми бачимо. Оскільки видиме світло не потрібно для побудови сцени, можна з легкістю спостерігати за об'єктами у повній темряві, а також у різних погодних умовах і навіть у тумані. Залежно від технології формування зображення є різні системи, які мають свої переваги й недоліки. Як показано на рис.1.1 [1], інфрачервоне випромінювання поділяється на NIR, SWIR, MWIR, LWIR та LWIR, що мають довжини хвиль від ближнього діапазону спектра до понаддовгохвильового відповідно. В інфрачервоних ОЕС використовується світло короткохвильового діапазону

(1,4 — 3) мкм, середнього діапазону (3 — 5) мкм та довгохвильового діапазону (8 — 14) мкм.

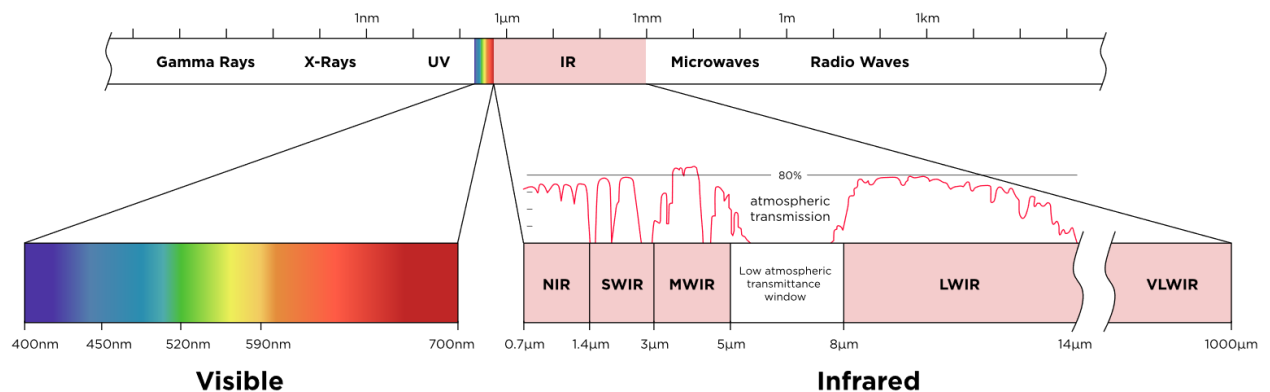


Рисунок 1.1 – ІЧ діапазони електромагнітного спектру

Камери MWIR і LWIR здатні визначати об'єкти на далеких дистанціях (іноді до 50 км) так як, люди, тварини і транспортні засоби, зазвичай, тепліше навколишнього середовища, що забезпечує високу контрастність. Теплова енергія може проходити через безліч видимих аерозолів, включаючи дим, пил, легкий туман, а також легке листя, що дає змогу тепловізорам дивитися через ці перешкоди. З метою безпеки термосенсиори, зазвичай, обмежені роздільною здатністю в 640×480 , що набагато нижче, ніж у більшості сучасних видимих сенсорів. Це може вплинути на ідентифікацію або розпізнавання об'єктів. Однак існують методи, які здатні збільшувати роздільну здатність зображення програмними шляхами.

Камери SWIR відрізняються від MWIR і LWIR тим, що вони дуже чутливі до світла й можуть приймати випромінювання від об'єктів, навіть у повній темряві завдяки явищу саява нічного неба, яке випромінює на 700 % більше світла, ніж світло зірок і складається в основному з довжин хвиль SWIR. Такі камери здатні бачити об'єкти з високим рівнем деталізації, навіть у безмісячні й беззоряні ночі, через туман і скло [2]. Однак, через високу продуктивність ця технологія захищена Міжнародними правилами про торгівлю зброєю, які накладають обмеження на те, куди ці продукти можуть бути експортовані і хто їх може купувати.

Взагалі, всі тепловізори можна розділити на охолоджувальні й неохолоджувальні. Тепловізійні камери з охолоджувальними сенсором володіють деякими перевагами в порівнянні з неохолоджувальним, однак вони дорожче. Неоохолоджувальні системи мають термосенсор, який називається мікроболометр. Він реагує на зміни опору при нагріванні або охолодженні. Тобто, коли випромінювання потрапляє на цей сенсор, він нагрівається і відповідно його опір змінюється. Потім зміна опору перетворюється в цифровий вигляд, калібрується за температурою і залежно від температури присвоюється значення кольору для кольорового уявлення або присвоюється значення різних рівнів градацій сірого для чорно-білого залежно від камери. Переваги таких тепловізорів це — низьке енергоспоживання, низька вартість, помірна чутливість і відсутність механічного охолодження. А до недоліків можна віднести низьку роздільну здатність та повільну частоту кадрів. Охолоджувальні камери працюють іншим чином. Вони збирають фотони інфрачервоної енергії, які проходять через оптику. Ці фотони перетворюються в електрони і зберігаються в конденсаторі. Після певного періоду часу заряд зчитується в цифровий відлік і також калібрується по температурі і, відповідно, присвоюється значення кольору або шкали сірого. Завдяки такому підходу маємо високу частоту кадрів, гарну чутливість, здатність бачити і вимірювати дуже маленькі об'єкти і кращу роздільну здатність.

При виконанні завдання розпізнавання та ідентифікації об'єктів, важливу роль відіграє роздільна здатність зображення. Існує безліч досліджень на тему того, як може впливати роздільна здатність на якість алгоритмів виявлення. Якщо проаналізувати дані, які показують вплив різних чинників на необхідну роздільну здатність для успішного детектування, то можна виділити такі фактори:

- Фактори навколишнього середовища (опаді, стан повітря, вологість, температура).

- Візуальні перешкоди, які не потребують розпізнання (нерівномірний фон, або фон, що складається з геометрично схожих предметів).
- Геометричні особливості об'єкту (розміри та розташування в просторі).

Деякі фактори, які вносять додаткові труднощі візуального сприйняття, представлені в табл. 1.1 [3].

Таблиця 1.1 – Фактори, що впливають на якість розпізнавання об'єктів

Фактор	Похибка виявлення, %	Коментар
Характеристика сцени	Від 50 до 250	Нерівномірний фон, щільність
Характеристика об'єкту	До 200	Розміри, розташування в просторі
Погодні умови	До 100	Опади, стан повітря, вологість, температура
Характеристика приладу	До 30	Роздільна здатність, різкість зображення, якість передачі.
Дисплей і спостерігач	До 100	Навколишні умови і кваліфікація спостерігача

Як видно з вищенаведеної таблиці, людський фактор дуже сильно може вплинути на розпізнавання і тому, на поточний момент розробляється велика кількість ОЕС в поєднанні з автоматизованими методами, які побудовані на нейромережевих технологіях.

1.2 Аналіз відомих технічних рішень побудови об'єкта дослідження за результатами проведеного патентного пошуку

При виконанні даної роботи був проведений патентний пошук та аналіз рішень ОЕС, які використовують два джерела відеопотоку в інфрачервоному

і видимому діапазонах, і які потенційно придатні для виявлення об'єктів завдяки своїй продуктивності. Для пошуку використовувалися інтернет-ресурси «Європейського патентного відомства», «Патентного відомства США», «Міжнародного патентного відомства» та «Роспатенту».

Патент Російської Федерації RU2563557. МПК: G01S 17/87, G03B 42/08. Винахід відноситься до технічних систем відеоспостереження для контролю обстановки на охоронній території. Система містить автоматизоване робоче місце оператора і оптико-електронний модуль, який включає в себе опорно-поворотний пристрій, телевізійну камеру, тепловізор і блок управління. В основу винаходу покладена технічна задача підвищення надійності виявлення і розпізнавання об'єктів при спостереженні їх цілодобово (вдень і вночі) в реальному часі.



Рисунок 1.2 – Структурна схема мультиспектральної системи оптико-електронного спостереження.

На рис.1.2 показана структурна схема мультиспектральної системи. Система спостереження складається з взаємозв'язаних пристроїв модуля оптико-електронного 1 і автоматизованого робочого місця (АРМ) оператора 10. В оптико-електронний модуль 1 входить оптико-електронний блок (ОЕБ) 2, опорно-поворотний пристрій (ОПП) 5 і блок управління 6. В оптико-електронному блоці 2 розміщені телевізійна камера (ТК) 3 і тепловізор 4. Оптичні системи 3 і 4 виконані за схемою зі змінною фокусною відстанню. До складу блоку управління 6 входить контролер 7, відеосервер 8 і пристрій коригування відеозображення 9. Команди управління від АРМ оператора 10 надходять у контролер 7, який після їхньої обробки передає по лінії зв'язку 17 в ОПП 5 сигнал управління приводом для повороту ОЕБ 2 в потрібному напрямку, а по лінії зв'язку 16 — сигнал управління наведенням ТК 3 на задану ділянку території. У даному патенті не вказано, який блок обробки зображення використовується і який формат має канал передачі (цифровий/аналоговий).

Патент США US20150324656. МПК: G06K 9/46, G06K 9/52, H04N 5/33, H04N 7/18, G01J 5/02, G01J 5/00. Спосіб і система відстеження людей у приміщенні шляхом отримання першої послідовності зображень за допомогою камери видимого світла, що має одну частоту кадрів, і другої послідовності зображень із тепловим інфрачервоним датчиком, що має іншу частоту кадрів. Дана система має кілька варіантів відстеження: з використанням тільки камер видимого світла, використанням тільки ІЧ-камер, з використанням ІЧ-камер і камер видимого світла разом. Рисунок 1.3 зображає блок-схему послідовності операцій способу виявлення та відстеження людей 101 в середовищі 102 відповідно до варіантів реалізації винаходу. Перша послідовність зображень 111 навколишнього середовища досягається камерою 110 видимого світла (RGB), що має першу частоту кадрів. Друга послідовність зображень 121 навколишнього середовища отримується інфрачервоним датчиком 120, що має другу частоту кадрів.

Люди відслідковуються в першій послідовності зображень 112, яка використовує фонову модель (M) 113 для отримання місцезнаходження людей 114. Теплі області 122 виявляються в другій послідовності зображень для отримання теплової локації 124. Місцезнаходження людей і теплові локації вирівнюються в просторово-часовому відношенні 131. Потім 140 перевіряє, чи збігаються вирівняні місцезнаходження й локації. Перевірені місцезнаходження оновлюють свої граничні рамки, щоби вони відповідали рамкам локації 150. Місцезнаходження, які не перевірені, видаляються 160. Це призводить до появи нового поточного набору місцезнаходжень 170. Потім фонову модель 180 оновлюється для кожного пікселя, що лежить поза поточним місцезнаходженням. Це дає нову модель 113. Одним з основних недоліків цієї системи є відстеження тільки людей і тільки в замкнутих приміщеннях. Також у патенті описано тільки методи відстеження й нічого не сказано про структуру пристрою.

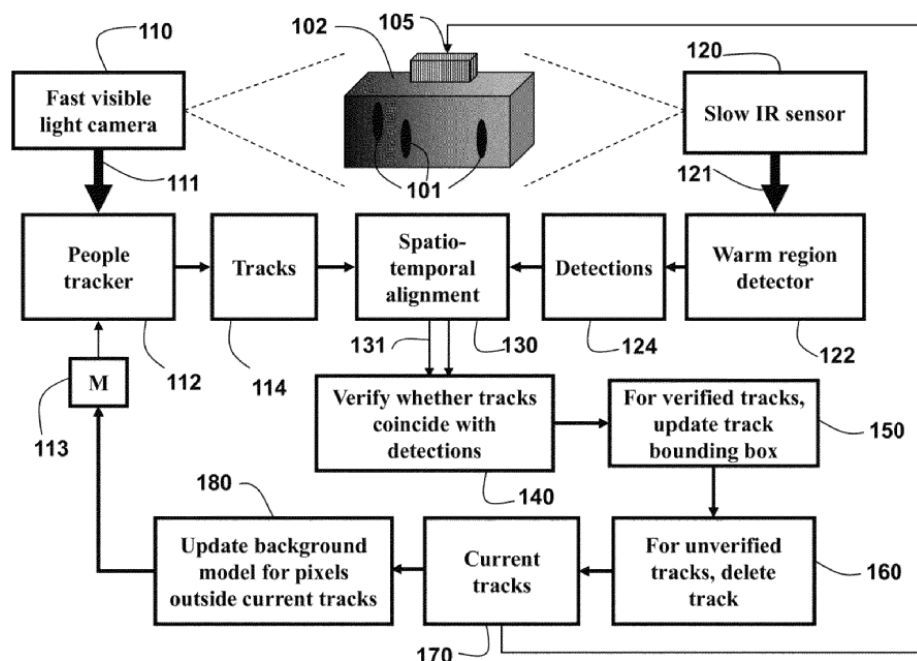


Рисунок 1.3 – Блок-схема методу відстеження людей

Патент Канади CA2658347. МПК: G08B 13/19, G08B 13/196. Даний винахід належить до систем безпеки і, зокрема, до виявлення і/або відстеження об'єктів у межах контрольованої області. Система безпеки складається з блоку спостереження, що складається з камери видимого світла та інфрачервоної камери. Пристрій може виявляти і/або відстежувати місце розташування й рух людей, тварин і машин. Виявляти та/або контролювати відкриття та закриття дверей, виявляти стан тривоги, повідомляти людей про умови тривоги або виконувати інші функції безпеки.

Рис.1.4 ілюструє схему блоку спостереження відповідно до варіанта реалізації винаходу, що розглядається.

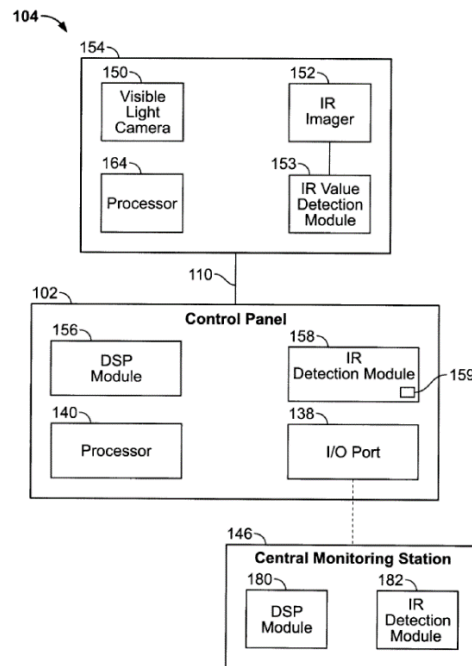


Рисунок 1.4 – Блок-схема вузла спостереження

Згідно з рис. 1.4 102 — системна панель керування має модуль цифрової обробки сигналів (DSP) 156 і модуль виявлення інфрачервоного випромінювання 158 для аналізу відео й даних з ПЧ камери. Процесор 140 передає відео та ПЧ-дані на центральну систему моніторингу 146 для аналізу і/або обробки модулем DSP 180 і модулем ПЧ-виявлення 182. Порт I/O 138 забезпечує інтерфейс зв'язку панелі 102 зі станцією моніторингу 146, яка

може бути підключена по бездротовому зв'язку, по телефонній лінії, локальної мережі (LAN), глобальної мережі (WAN), інтернету і под. Блок спостереження 104 містить камеру видимого світла 150 та ІЧ-сенсор 152, що міститься всередині корпусу 154. Модуль виявлення 153 використовується для виявлення рівня інфрачервоного випромінювання, який вимірює ІЧ-камера 152. Процесор 164 передає відеодані й ІЧ-дані на панель керування 102 по мережі 110. Даний патент є одним найближчим рішенням до бажаного пристрою. Він має дві камери — ІЧ-спектра і видимого. Дані з камер надходять у вигляді цифрового формату й система може відстежувати, як живі, так і неістотні об'єкти. Але в патенті зазначено, що для обробки відеопотоку використовується процесор цифрової обробки сигналів (DSP), що може викликати труднощі при реалізації сучасних методів виявлення об'єктів із використанням згорткових мереж.

Патент Російської Федерації RU2701177. МПК: G01S 17/46. Винахід належить до галузі пасивної локації, який використовується для оптичного виявлення, супроводу й розпізнавання наземних і повітряних об'єктів (НПОБ). Завданням винаходу є розширення функціональних можливості ОЕК з обслуговування безлічі рухомих НПОБ та ідентифікації їхнього типу. Технічним результатом, яке забезпечує рішення цієї задачі, є підвищення продуктивності ОЕК. Згідно Рис.1.5 ОЕК включає оглядовий тепловізор (ОТ) 1 для пасивного виявлення НПОБ і телевізійну оптичну систему (ТОС) 2 для точного вимірювання координат, дальності, автоматичного захоплення на супровід і ідентифікації типу НПОБ. По сигнальних виходах і командах управління оптичних приймачів (ОП) тепловізор 1 з'єднаний з ЕОМ 6 управління оглядом і обробкою тепловізійних сигналів ОТ 1. Виходи приймачів 2.1.1.1 ОП1, 2.1.1.2 ОП2 і 2.1.1.3 ОП3 з'єднані з входами багатоканального блоку 4 модулів 4.1, 4.3 автозахоплення й ідентифікації (МАІ). Недоліком даної системи є коштовна реалізація всього комплексу. Кожен модуль 4.1–4.3 виконаний у вигляді комп'ютера, який містить

процесор «Intel i5 8600K», графічний прискорювач «NVIDIA 2080 RTX, 8 Гб, 2944 ядра CUDA», оперативну пам'ять ємністю не менше 16 Гб, внутрішню пам'ять «128 Гб SSD + 2 Тб HDD». ЕОМ 3 супроводу й управління містить ввідно-вивідний пристрій «Ethernet» 2×10 Гбіт, цифровий дискримінатор управління ОПУ, процесор «Intel Xeon» з частотою 2,4 ГГц, графічний контролер «NVIDIA GT 730», мережеву карту з продуктивністю не менше 10 Гб/с, оперативну пам'ять ємністю не менше 16 Гб і внутрішню пам'ять «SSD» і/або «HDD» місткістю не менше 1000 Гб.

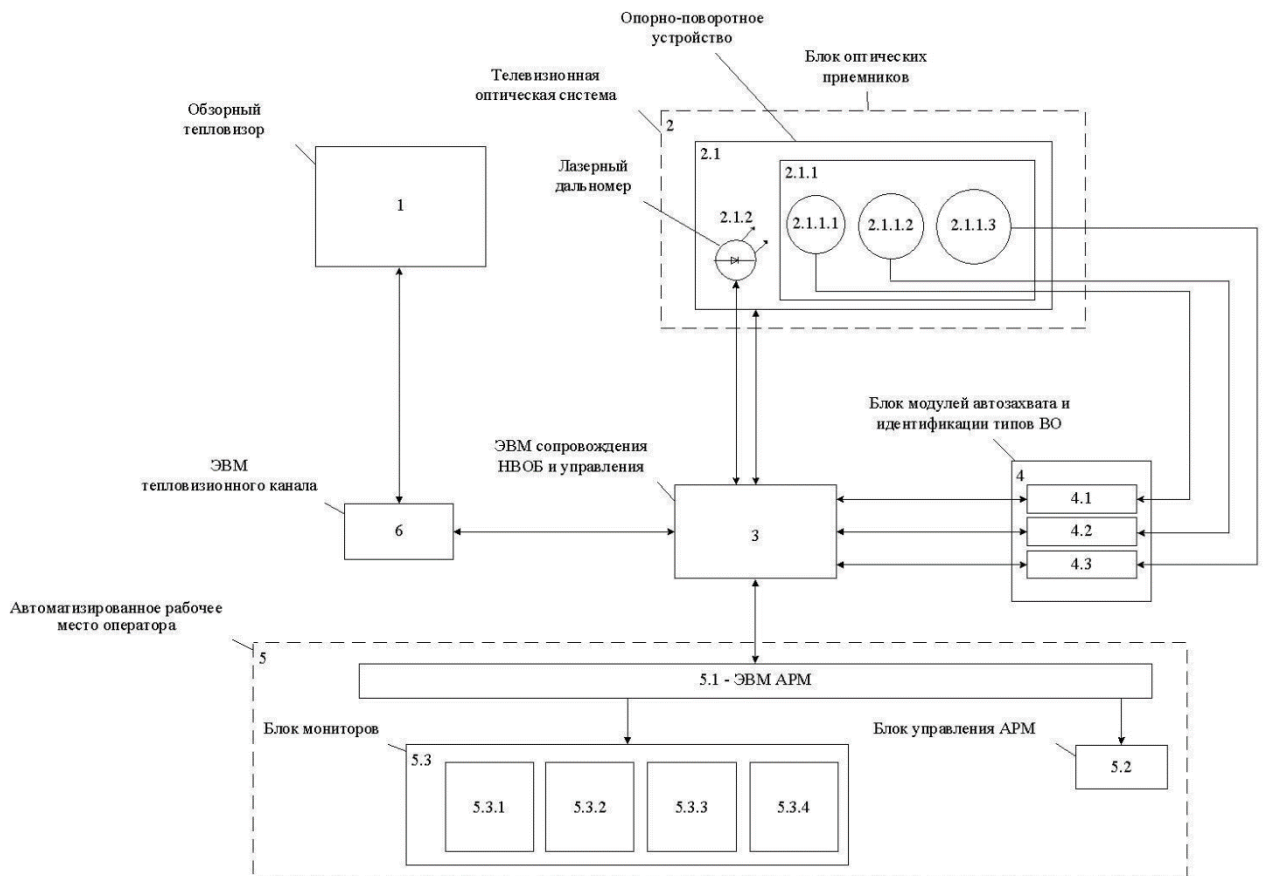


Рисунок 1.5 – Блок схема оптико-электронного комплекса

ЕОМ 6 ОТ містить процесор «Intel Core i7» з частотою 2.8 ГГц, графічний прискорювач «NVIDIA GeForce GTX 1050 Ti» з блоком пам'яті не менше 4 Гб, оперативну пам'ять «DDR4» ємністю 4-16 Гб, внутрішню пам'ять «SSD і HDD », ємністю 128 Гб і 2 Тб відповідно.

1.3 Порівняльний аналіз сучасних методів згорткових нейронних мереж розпізнавання об'єктів в режимі реального часу та постановка завдання дослідження

Виявлення об'єктів — це технологія, яка пов'язана з комп'ютерним зором і основне завдання якої, виявляти екземпляри семантичних об'єктів певного класу (наприклад людей, транспортні засоби, тварин) на відеопослідовностях або статичних зображеннях. Завдяки цій технології зараз можливе безпілотне водіння, відеоспостереження без оператора, виявлення обличчя та інші важливі галузі. Завдяки недавнім досягненням в XXI столітті з'явилося багато інноваційних та креативних методологій, які можуть бути, або з високими характеристиками, але ресурсоємні для використання в додатках реального часу, або навпаки менш точними, але давати нагоду користувачам використовувати виявлення об'єктів у компактних модульних структурах і навіть на мобільних пристроях.

У даній роботі не будуть розглянуті моделі з архітектурою Regions With Convolution Neural Networks (R-CNNs) та їхні похідні, які чудово справляються з завданням розпізнавання в автономному режимі, але абсолютно не підходять для реалізації в реальному часі. Такі алгоритми використовують двоетапний підхід — спочатку визначають області, де об'єкти повинні бути знайдені, а потім виявляють об'єкти тільки в цих, окремих регіонах, використовуючи згорткові нейронні мережі. З іншого боку, є алгоритми, які використовують повністю згортковий підхід, наприклад — Single Shot Detector (SSD), при якому мережа може знаходити всі об'єкти на зображенні за один прохід. Алгоритми з регіонами мають кращу точність, але повільніше в роботі, у той час, як алгоритми «одиночного пострілу» більш ефективні й мають таку ж високу точність. Отже, зосередимося в цьому розділі на цих алгоритмах.

1.3.1 Single Shot Detector (SSD)

SSD [4] складається з двох компонентів: базова частина й головна частина. Базова частина являє собою попередньо навчену мережу класифікації зображень, яка виділяє ознаки. В основі цієї моделі лежить ResNet-подібна нейромережа, навчена на певному наборі даних, з якої був видалений повнозв'язний рівень класифікації. Таким способом, залишається глибока нейронна мережа, яка здатна витягнути смислове значення з вхідного зображення, зберігаючи просторову структуру, хоча і з меншою роздільною здатністю. Головна частина SSD — це один або кілька згортальних шарів, доданих до базової частини, а вихідні дані інтерпретуються як обмежувальні рамки і класи об'єктів у просторовому розташуванні активацій кінцевих шарів. На рис.1.6 [5] перші кілька шарів (білі прямокутники) є базовою частиною, а останні кілька шарів (сині прямокутники) відображають головну частину SSD.

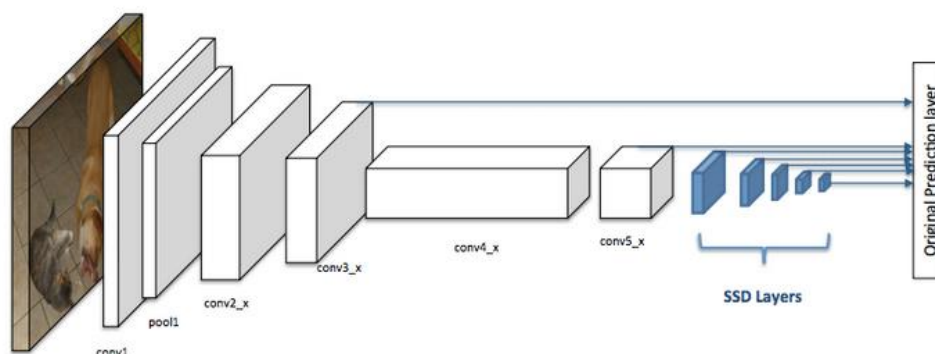


Рисунок 1.6 – Архітектура згорткової нейронної мережі з детектором SSD

Замість використання ковзаючого вікна (як в R-CCN), SSD ділить зображення з використанням сітки і кожна клітинка сітки відповідає за виявлення об'єктів у цій області зображення. Виявлення об'єктів означає прогнозування класу й місця розташування об'єкта в цьому регіоні. Якщо об'єкт відсутній, він розглядається як фоновий клас, а місце розташування ігнорується. SSD використовує менші шари для виявлення об'єктів більшого розміру. Наприклад, карти ознак 4×4 використовуються для об'єктів більшого масштабу, а 8×8 — для меншого (рис.1.7).

Якщо в одній комірці сітки кілька об'єктів або необхідно виявити кілька образів різної форми, SSD має рамки прив'язки (anchor box) і рецептивне поле (receptive field). Кожній клітинці сітки в SSD може бути призначено декілька точок прив'язки. Ці поля попередньо визначені, і кожне з них відповідає за розмір і форму в клітинці. Наприклад, басейн на рис.1.8 [5] відповідає зеленому полю прив'язки, у той час, як будівля відповідає фіолетовому. По суті, прямокутник прив'язки, який найбільше перекриває об'єкт, відповідає за прогнозування класу цього об'єкта і його місця розташування. Ця властивість використовується для навчання мережі і для прогнозування виявлених об'єктів і їх розташування. Кожна рамка прив'язки визначається співвідношенням сторін і рівнем масштабування.

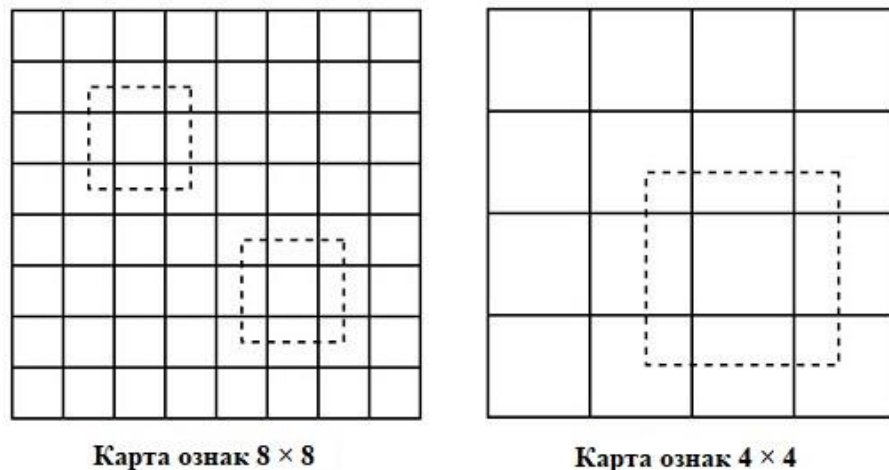


Рисунок 1.7 – Карти ознак меншого (зліва) і більшого (справа) масштабу

Об'єкти мають різну форму. Деякі можуть бути довшими, а інші ширше. Архітектура SSD дозволяє різним співвідношенням сторін областей прив'язки це враховувати.

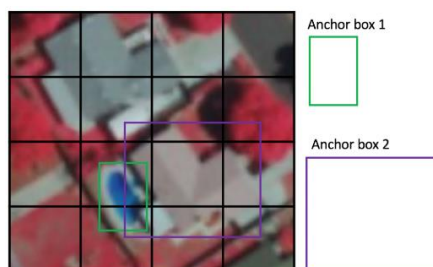


Рисунок 1.8 – Приклад застосування рамок прив'язки

Через операцію згортки об'єкти на різних шарах представляють різні розміри області на вхідному зображенні. У міру того, як вона поглиблюється, розмір, представлений функцією, збільшується. У прикладі на рис.1.9 рецептивне поле починається з нижнього шару (5x5) і потім застосовуючи операцію згортки, переходить до середнього прошарку (3x3), де одна комірка (зелений піксель) представляє область 3x3 вхідного нижнього шару. Наступним кроком, застосовуючи згортку до середнього шару отримаємо верхній шар (2x2), де кожен елемент відповідає області 7x7 на вхідному зображенні. Рецептивне поле є основною ідеєю архітектури SSD, оскільки воно дозволяє виявляти об'єкти в різних масштабах і виводити більш щільну граничну рамку.

Отже, основні особливості SSD:

- SSD здійснює багато прогнозів і має гарне покриття на зображенні.
- Через видалення регіонів і використання відеопослідовностей із більш низькою роздільною здатністю модель може працювати в режимі реального часу.
- Продуктивність SSD512 з використанням ЗНН із низьким обсягом пам'яті HarDNet85 має наступні параметри: показник точності $mAP@0.5 = 35,1$; значення FPS = 34.

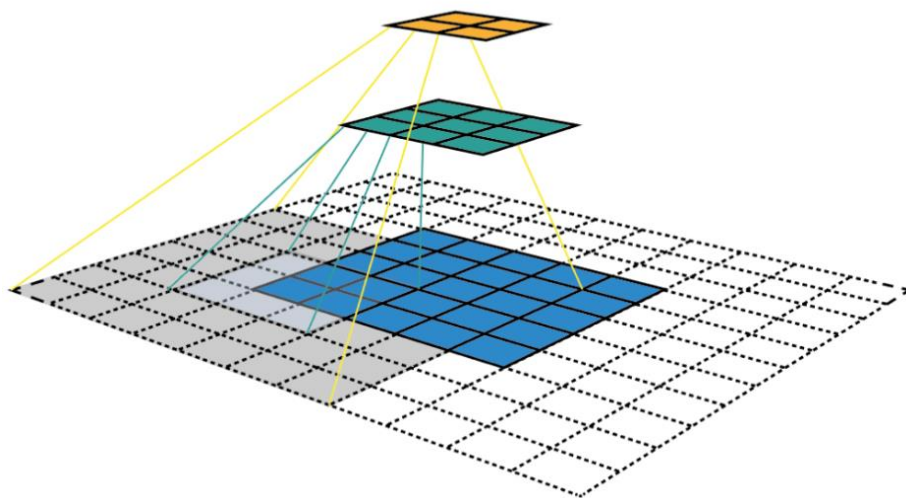


Рисунок 1.9 – Візуалізація карт ознак і рецептивного поля

1.3.2 Feature Pyramid Network (FPN)

Один зі способів виявлення об'єктів розроблено спільними зусиллями дослідників Корнельського Університету і Facebook AI. В основі архітектури FPN [6] використовуються «ознакові піраміди», які дозволяють виявляти об'єкти різного масштабу з кращою точністю і схожою швидкістю, у порівнянні з попередньою архітектурою SSD. SSD працює за схожим принципом виявлення, який складається з декількох карт ознак і рецептивного поля, проте нижні шари не використовуються для виявлення. Хоча, ці шари мають високу роздільну здатність, SSD їх не враховує через втрату продуктивності. Тому в SSD використовується тільки верхні рівні для виявлення і, отже, вона працює набагато гірше для невеликих об'єктів. FPN створює архітектуру з багатою семантикою на всіх рівнях, оскільки вона поєднує змістовно сильні функції низької роздільної здатності зі змістовно слабкими функціями високої здатності. Це досягається завдяки створенню спадного (top-down) шляху з горизонтальними зв'язками (lateral connections) до висхідних (bottom-up) згорткових шарів (рис.1.10) [6].

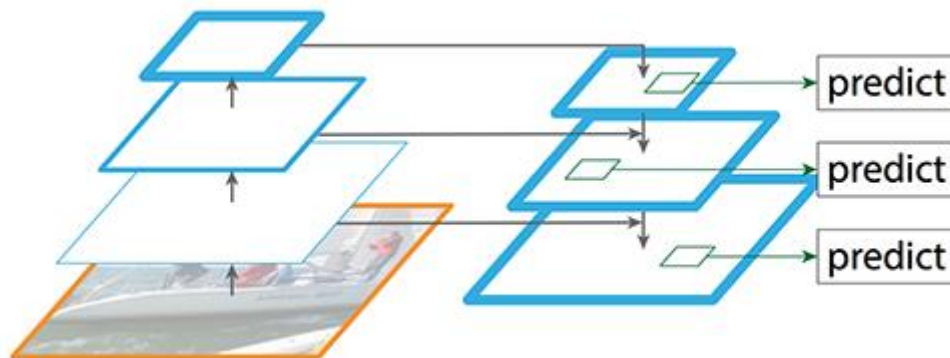


Рисунок 1.10 – Спадна архітектура FPN з незалежними прогнозами на всіх шарах

Висхідний шлях (bottom-up pathway) – це базова частина (наприклад ResNet), яка розраховує карти ознак у різних масштабах незалежно від розміру вхідного зображення.

Спадний шлях (top-down pathway) збільшує вибірку просторових карт, починаючи з більш високих рівнів піраміди.

Горизонтальні зв'язки (lateral connections) об'єднують спадні і висхідні шари з однаковим просторовим розміром.

FPN найчастіше застосовується у зв'язку з іншими детекторами й архітектурами. Наприклад, з RPN [7], Mask R-CNN [8], AmoebaNet [9] та іншими. Завдяки такому підходу продуктивність вихідних архітектур може підвищуватися більше, ніж у 2 рази. Mask R-CNN на основі FPN досягає часу виведення 0,148 секунди на зображення для ResNet-50, з іншого боку, базова Mask R-CNN без FPN обробляє те саме зображення за 0,32 секунди [8].

Основні особливості FPN:

- У FPN використовується спадний шлях, який відновлює роздільну здатність завдяки багатій семантичній інформації.
- З горизонтальними зв'язками додається точніша просторова інформація про об'єкт.
- Спадний шлях і горизонтальні зв'язки підвищують точність на 8 пунктів у наборі даних MS COCO. Для невеликих об'єктів поліпшення збільшується на 12,9 балів [6].

1.3.3 RetinaNet

RetinaNet [10] — одна з кращих одноетапних моделей виявлення, що добре зарекомендувала себе при роботі зі щільними і дрібними об'єктами. З цієї причини вона стала популярною моделлю, що використовується з аерофотознімками й супутниковими зображеннями. RetinaNet сформована через об'єднання наявних моделей — Feature Pyramid Network (FPN) і функції помилки Focal Loss (FL). Окрім вищезазначених моделей RetinaNet має підмережі класифікації й регресії.

FL є покращеною функцією помилки в порівнянні з помилкою на основі перехресної ентропії і вводиться для розв'язання проблеми дисбалансу класів в одноетапних моделях виявлення об'єктів. Ця проблема дисбалансу виникає через щільну вибірку рамок прив'язки. Наприклад, SSD повторно робить вибірку співвідношення класу об'єкта і фону під час навчання, для того щоби не перевантажувати модель фоном зображення. FL використовує інший підхід, щоби зменшити втрати для добре навченого класу. Всякий раз, коли модель добре розпізнає фон, вона зменшує його втрати і знову акцентує увагу на навчанні в класі об'єкта. На рис.1.11 приведена архітектура Retina Net, яка базується на FPN і ResNet і використовує функцію Focal Loss.

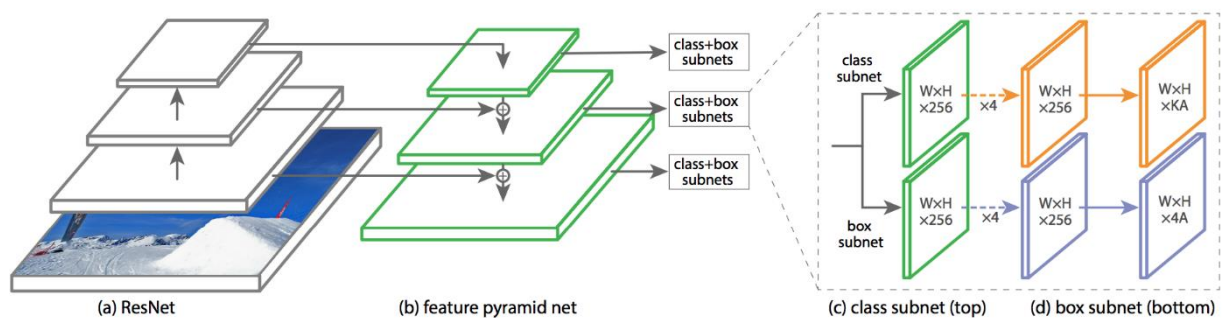


Рисунок 1.11 – Архітектура Retina Net

Перші дві піраміди – це вже розглянуті висхідний шлях (a) і спадний шлях (b) з архітектури FPN. Класифікаційна підмережа (c) прогнозує ймовірність присутності об'єкта в кожному просторовому місцезнаходженні для кожної рамки прив'язки і класу об'єкта. Підмережа регресії (d) – це повторне зміщення для граничних рамок (bounding boxes) з рамок прив'язки для кожного об'єкта достовірності [10].

Основні особливості RetinaNET:

- Об'єднує дві архітектури FPN + ResNet і завдяки спеціальній функції помилки (focal loss) має більш високу точність.
- RetinaNet перевершує по частоті кадрів попередні SSD і FPN.

1.3.4 You Only Look Once (YOLO)

Перша версія YOLO [11] з'явилася в галузі комп'ютерного зору у 2015 році за авторством Джозефа Редмона і відразу ж привернула багато уваги дослідників цієї сфери. YOLO є найпопулярнішою архітектурою, яка використовується для виявлення об'єктів у реальному часі й перша модель, яка реалізована на мобільному пристрої. Алгоритм «дивиться лише раз» застосовується до зображення в тому сенсі, що для прогнозування потрібно тільки один прохід прямого розповсюдження через нейронну мережу.

Розглянемо принцип роботи YOLO (рис.1.12). Алгоритм починається з того, що мережа ділить зображення на своєрідну сітку. Залежно від роздільної здатності, сітка має певну кількість клітинок. Потім вона прогнозує відповідну кількість граничних рамок (bounding boxes) для кожної ділянки. Та прогнозує ймовірності того, що в цих рамках є якісь бажаний об'єкт. Кожен граничний блок містить 5 елементів: ширина w , висота h , зміщення до відповідної комірки x і y , а також показник достовірності блоку. Показник достовірності говорить, наскільки мережа впевнена, що передбачуване поле, або рамка насправді містить якийсь об'єкт. На середньому зображенні рис.1.12 (б) видно, як нейронна мережа промалювала всі рамки, які мають значення показнику достовірності. Чим більше цей показник, тим товщі рамки.

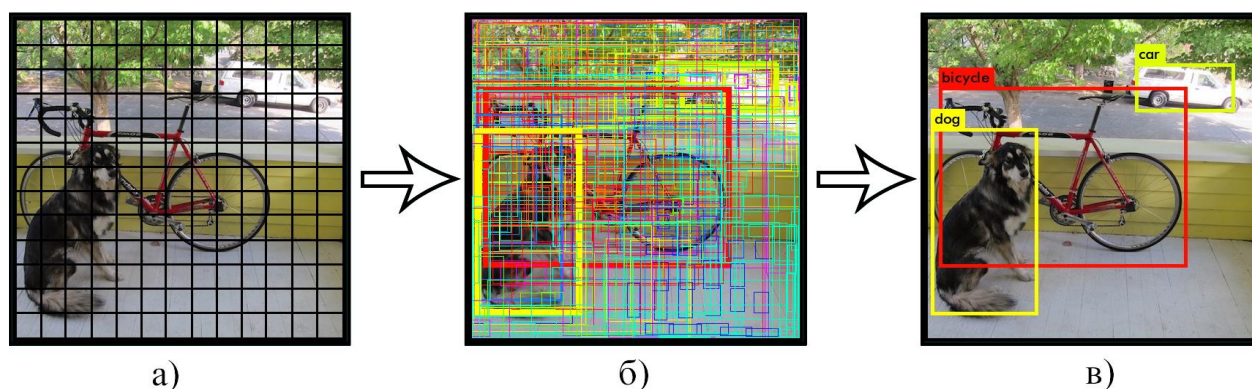


Рисунок 1.12 – Принцип роботи архітектури YOLO

Далі починається етап класифікації і мережа для кожного граничного вікна передбачає клас. Показник достовірності та значення прогнозування класу об'єднуються в один загальний бал, який говорить про ймовірність того, що ця рамка містить конкретний тип об'єкта. YOLO може передбачити кілька граничних рамок на одну клітинку сітки. За допомогою функції помилки ЗНМ знаходить єдине справжнє значення і відкидає рамки, які мають менше значення показника. YOLO використовує суму квадратів помилок між прогнозами й істинною величиною для розрахунку втрат. Функція втрат складається з:

- класифікації втрат;
- локалізації втрат (помилки між передбаченою і істинною граничною рамкою);
- втрата довіри (предметність рамки).

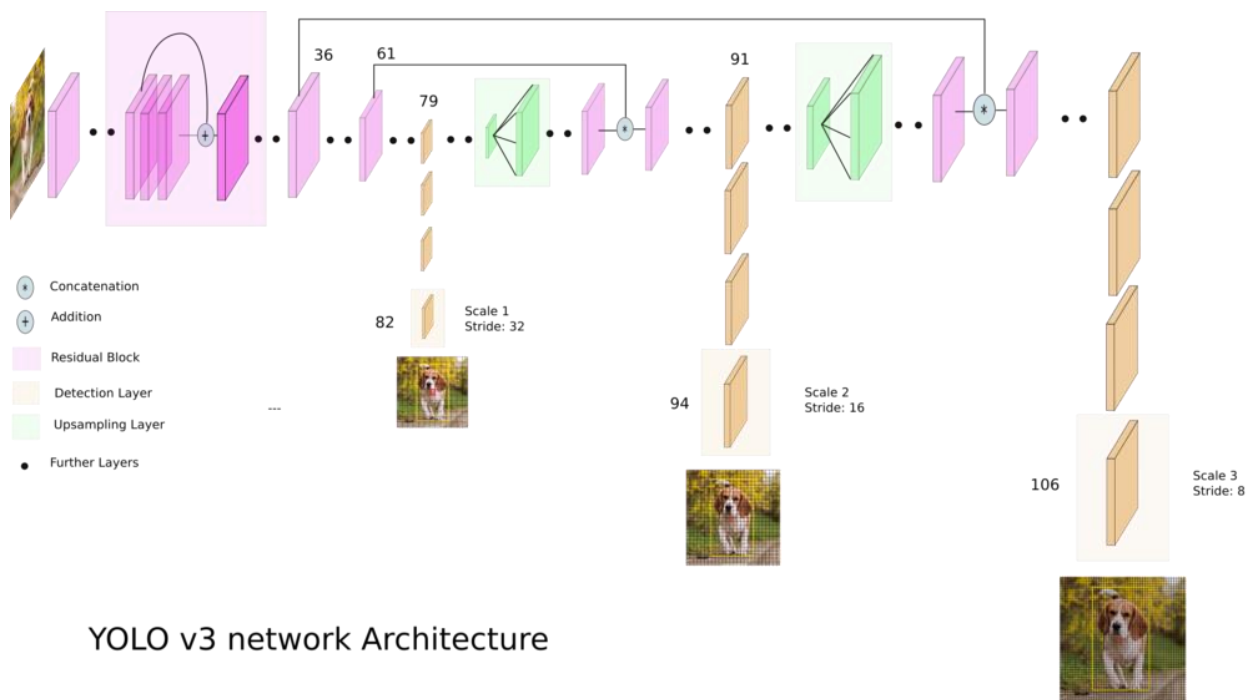


Рисунок 1.13 – Архітектура YOLOv3

Існують поліпшенні версії цієї моделі, а саме, YOLOv3, YOLOv3-tiny і найсучасніші YOLOv4 та YOLOv5. На рис.1.13 показано архітектуру YOLOv3 [12]. Вона використовує «ознакові піраміди» FPN, тобто складається з більшої кількості згорткових шарів і краще розпізнає невеликі

об'єкти. YOLOv3 робить прогнози в трьох різних масштабах. На кожному шарі вона будує сітку з різним кроком. Перший шар має великий крок 32 і, відповідно, на ньому краще виявляються великі об'єкти, а останній має крок 8, що виявляє, навпаки, дрібні об'єкти. Прогноз починається з останнього шару карти ознак. Потім він повертається на два шари назад і збільшує карту на два. Наступним кроком виділяється карта ознак із більш високою роздільною здатністю й об'єднується з уже збільшеною картою ознак, використовуючи поелементне додавання.

Потім мережа застосовує згорткові фільтри до об'єднаної карти, щоби передбачити другий набір прогнозів. Ця дія триває доти, поки одержуваний шар карти об'єктів не досягне хорошої структурної (семантичної) і просторової інформації високого рівня з гарною роздільною здатністю для розташування об'єктів.

У третій версії YOLO використовується новий 53-шаровий блок виділення ознак Darknet-53. Він складається з фільтрів 3×3 і 1×1 зі східчастими з'єднаннями, як в ResNet. Darknet-53 має менше операцій із плаваючою комою (BFLOP), ніж ResNet-152, але досягає тієї ж точності класифікації у 2 рази швидше.

Для порівняння YOLO з іншими сучасними архітектурами наведена табл.1.2, в якій представлено архітектури, що використовуються в реальному часі. Можна зробити висновок, що моделі YOLO мають оптимальне співвідношення швидкості й точності для вбудованих пристроїв.

Таблиця 1.2 – Порівняння YOLO з іншими архітектурами

<i>Модель</i>	<i>Точність, mAP@0.5</i>	<i>BFLOPS</i>	<i>Швидкість, FPS</i>
SSD512- HarDNet85	35.1	-	34
NAS-FPN AmoebaNet	48.3	-	36

Продовження табл.1.2

<i>Модель</i>	<i>Точність, mAP@0.5</i>	<i>BFLOPS</i>	<i>Швидкість, FPS</i>
RetinaNet (SpineNet)	45.3	-	39
YOLOv3-320	51.5	38.97	45
YOLOv3-tiny	33.1	5.56	220
YOLOv4-512	43.5	-	83

Слід враховувати, що ці дані отримані з використанням однакових графічних процесорів і навчанні на наборі Microsoft COCO — великомасштабному наборі даних для виявлення та сегментації. Основні особливості MS COCO [13]:

- 330 000 зображень (> 200 000 марковані)
- 1,5 мільйона примірників об'єкта
- 80 категорій об'єктів
- 91 категорій речей
- 5 підписів на одному зображенні
- 250 000 людей з точками захоплення

Для нашого завдання необхідно розпізнавати об'єкти в інфрачервоному діапазоні, відповідно навчальний набір даних повинен складатися з теплових зображень. У вільному доступі є набір даних компанії FLIR, про який буде йти мова докладніше в розд. 3. Ефективність навчання ГНН, яку запропонувала компанія FLIR, є недостатньою для застосування в реальних прикладних задачах, отже, стоїть завдання поліпшити показники навчання до більш прийнятних.

Висновки за розд. 1

1. Проведено аналіз факторів, які можуть тією чи іншою мірою і вплинути на виявлення й розпізнавання об'єктів в оптико-електронних системах, що працюють у видимому та інфрачервоному діапазонах. Виявилось, що найбільший негативний внесок у розпізнавання вносять візуальні перешкоди, нерівномірний фон, погодні умови й людський фактор спостерігача.

2. Проведено патентний пошук технічних рішень побудови оптико-електронних комплексів, які призначені для автоматичного виявлення об'єктів без втручання людини. Порівняльний аналіз дозволяє зробити висновок, що всі пристрої відрізняються відсутністю завадостійкого цифрового швидкісного каналу передачі даних.

3. Проведено пошук сучасних одноетапних підходів виявлення об'єктів, які використовуються в реальному часі і придатні для реалізації на компактних модульних структурах або вбудованих системах. Аналіз показав, що YOLO — найефективніша архітектура з погляду продуктивності, точності і швидкості обробки на поточний момент, що визначило її вибір в якості базової для подальшого удосконалення в роботі.

4. Сформульовано постановку завдання дослідження, яке полягає в збільшенні ефективності навчання згорткової нейронної мережі YOLO на основі набору даних із тепловізійних зображень.

РОЗДІЛ 2. ДОСЛІДЖЕННЯ РІЗНОВИДІВ АРХІТЕКТУР YOLO ТА ОБҐРУНТУВАННЯ ВИБОРУ МОДЕЛІ НЕЙРОННОЇ МЕРЕЖІ

2.1 Архітектура і конфігурація YOLOv4

У 2020 році утворилася жорстка конкуренція між моделями для виявлення об'єктів, про які буде розказано нижче. Однак, версія YOLOv4, як стверджують її засновники, збрала в собі всі останні досягнення науки в цій галузі. Вона досягає точності 43,5 % mAP (65,7 % mAP₅₀) для MS COCO зі швидкістю приблизно 65 FPS на графічному процесорі NVIDIA Tesla V100 [14]. Однак, висока точність не єдиний головний параметр у виявленні. Для досягнення поставленої мети важлива плавність роботи на периферійних пристроях і можливість обробляти відео в реальному часі без високовартісного обладнання.

Покращення процесу навчання можна розділити на дві групи. Ті, які не впливають на швидкість, але покращують точність (наприклад збільшення кількості даних, функції втрат, незбалансованість класів, категорювання класів із прив'язаною оцінкою ймовірності). І група, яка впливає на час швидкості виведення (наприклад, збільшення рецептивного поля, використання «візуальної уваги» (visual attention), інтеграція модуля «пропуску з'єднань» (skip connection) і FPN, а також ключовий крок постобробки — придушення немаксимумів (NMS)). Отже, розглянемо більш детально ці опції.

2.1.1 Базова частина

Для підвищення точності можна розробити більш глибоку мережу — розширити рецептивне поле й, таким способом, підвищити комплексність моделі. А щоби полегшити навчання, можна застосувати пропуск з'єднань. І

завдяки сильно взаємозалежних шарів, можна ще більше розширити загальний задум.

Розглянемо модуль *DenseBlock* (рис.2.1), який використовується в YOLOv4 для об'єднання шарів один з одним зіставляючи їхні розміри карти ознак. Dense layer складається з декількох згорткових шарів x_i і прошарків пакетної нормалізації H_i [15].

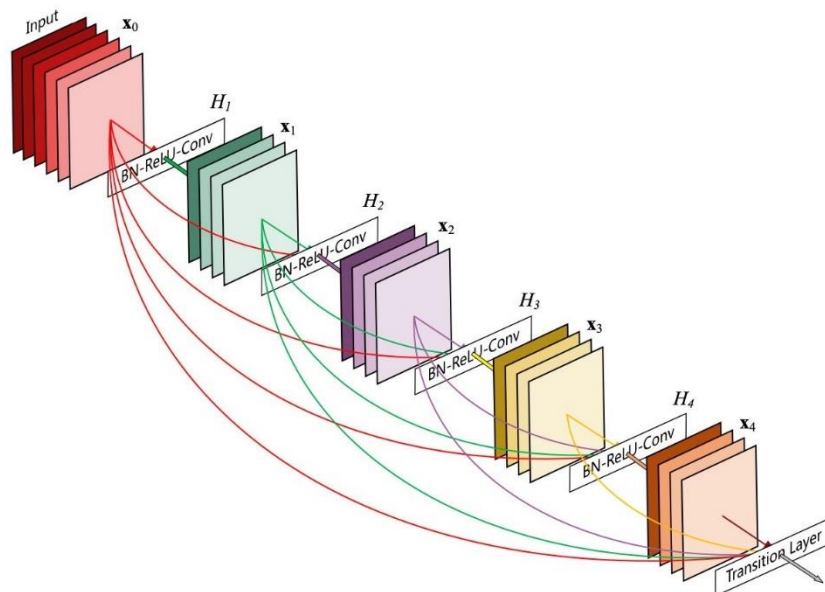


Рисунок 2.1 – П'ятишаровий модуль DenseBlock

H_i приймає на вхід оригінал у якості вхідних даних і вихід усіх попередніх шарів, тобто x_0, x_1, \dots, x_{i-1} , що дає нагоду зберегти більше інформації. Кожен попередній шар H_i виводить чотири карти ознак. Отже, на кожному шарі кількість карт ознак збільшується на чотири, що дає збільшення швидкості росту.

Наступний модуль базової частини YOLOv4 – *CSPNet* [16]. Він відокремлює вхідні карти ознак DenseBlock на дві частини. Перша

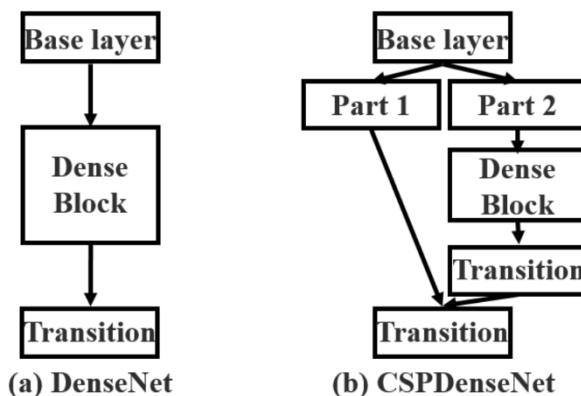


Рисунок 2.2 – Різні стратегії об'єднання карт функцій в CSPNet.

частина x_0' обходить DenseBlock і стає частиною вхідних даних для наступного перехідного шару. Друга частина x_0'' буде проходити через DenseBlock, як показано на рис.2.2 (b). Завдяки цьому модулю, вхід розділяється на дві частини й тільки один проходить через DenseBlock. Це знижує обчислювальну складність.

В YOLOv4 застосовується технологія об'єднання з'єднань CSP і Darknet-53 в якості основи базової частини. Архітектура Darknet-53, яка використовувалася в третій версії YOLO зображена на рис.2.3 [12].

	Type	Filters	Size	Output
	Convolutional	32	3×3	256×256
	Convolutional	64	$3 \times 3 / 2$	128×128
1x	Convolutional	32	1×1	128×128
	Convolutional	64	3×3	
	Residual			
	Convolutional	128	$3 \times 3 / 2$	64×64
2x	Convolutional	64	1×1	64×64
	Convolutional	128	3×3	
	Residual			
	Convolutional	256	$3 \times 3 / 2$	32×32
8x	Convolutional	128	1×1	32×32
	Convolutional	256	3×3	
	Residual			
	Convolutional	512	$3 \times 3 / 2$	16×16
8x	Convolutional	256	1×1	16×16
	Convolutional	512	3×3	
	Residual			
	Convolutional	1024	$3 \times 3 / 2$	8×8
4x	Convolutional	512	1×1	8×8
	Convolutional	1024	3×3	
	Residual			
	Avgpool		Global	
	Connected		1000	
	Softmax			

Рисунок 2.3 – Архітектура Darknet-53

Нова основа називається *CSPDarknet53*. Вона має більш високу точність виявлення об'єктів у порівнянні зі структурами на основі ResNet, хоча вони краще працюють із класифікацією об'єктів. Але точність класифікації CSPDarknet53 може бути поліпшена за допомогою Mish і інших методів, що обговорюються нижче.

2.1.2 Шийна частина

У детекторах об'єктів базова частина відповідає за виділення ознак, а головна частина – безпосередньо за виявлення об'єктів. Щоби примножити інформацію, що надходить у головну частину, сусідні карти ознак або поелементно додаються або конкатенуються. Частина, яка об'єднує карти ознак знизу догори (просторова інформація) і згори донизу (семантична інформація) називається шийною.

Для виділення ознак в YOLO використовується підхід схожий на FPN, який був розглянутий у першому розділі, для прогнозування виявлення об'єктів у різних масштабах.

Для усунення обмеження розміру вхідного зображення та детектування об'єктів різного масштабу використовується технологія Spatial Pyramid Pooling (SPP) [17]. Вона замінює останній шар агрегування шаром об'єднання «просторової піраміди» (рис.2.4). Для збереження вихідного просторового виміру в YOLO, SPP трохи змінена. Розмір ковзаючого вікна змінений на непарні значення на відміну від парних у стандартній версії SPP. Просторовий вимір залишається незмінним. На виході, карти ознак, із різними розмірами ядра, конкатенуються разом. На рис. 2.5, нижче, показано, як SPP інтегрована в YOLO.

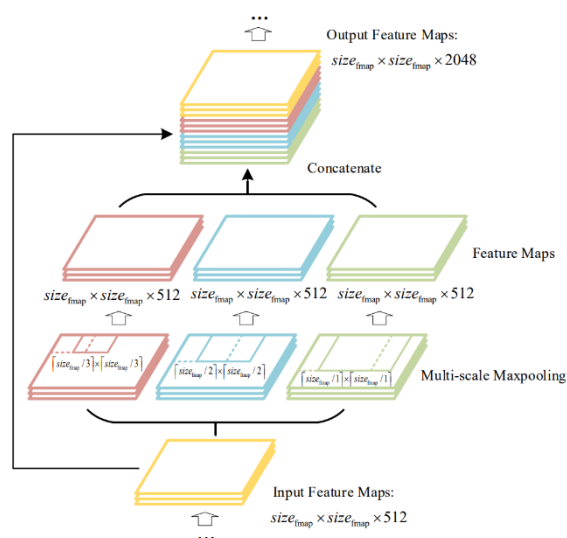


Рисунок 2.4 – Покращена технологія об'єднання SPP для YOLO.

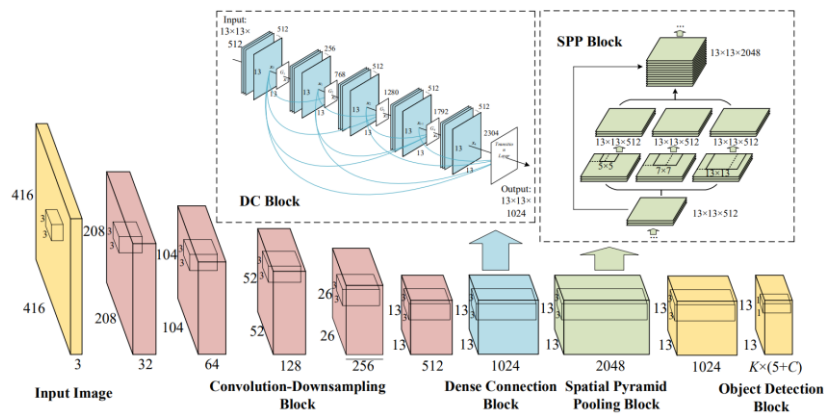


Рисунок 2.5 – SPP інтегрована в YOLO

Взаємозв'язки між шарами в сучасних моделях розпізнавання стають складніше й більш змішаними. Якщо раніше кожен шар приймав вхідні дані з попереднього шару, то зараз кожен шар пов'язаний з усіма попередніми, як наприклад, в DenseBlock на рис.2.1. Відповідно по мірі просування вправо локалізована інформація, яка може знадобитися для точного передбачення класу, може бути втрачена і виникає питання транспортування інформації між шарами. Нижче на рис.2.6 наведена технологія *Path Aggregation Network* (PAN).

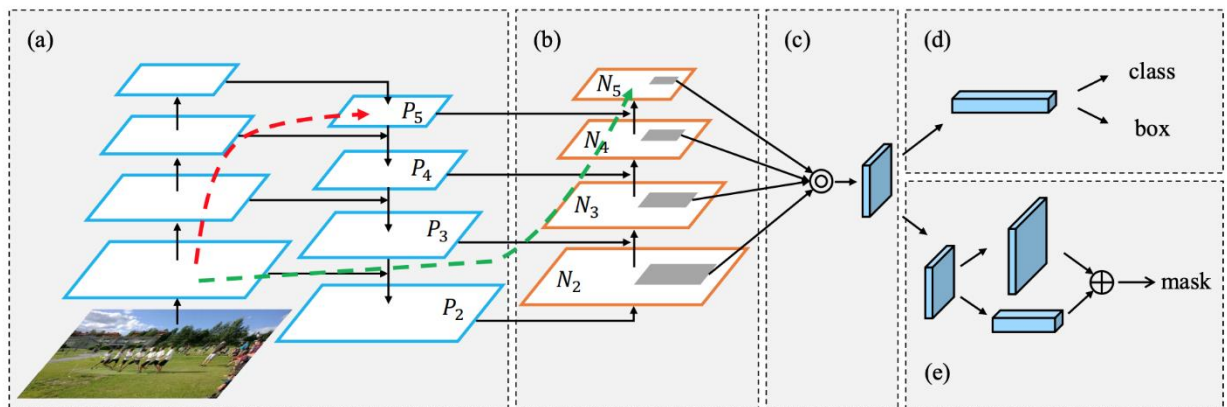


Рисунок 2.6 – Path Aggregation Network

У FPN локалізована просторова інформація переміщується вгору за червоною стрілкою. Хоча це не ясно показано на діаграмі, червоний шлях проходить приблизно через 100+ шарів. PAN пропонує скорочений (зелений) шлях, який займає орієнтовно 10 шарів, щоби перейти до верхнього шару N_5 .

Однак, замість того, щоби підсумувати сусідні шари, карти ознак в YOLOv4 конкатенуються (рис.2.7) [14].

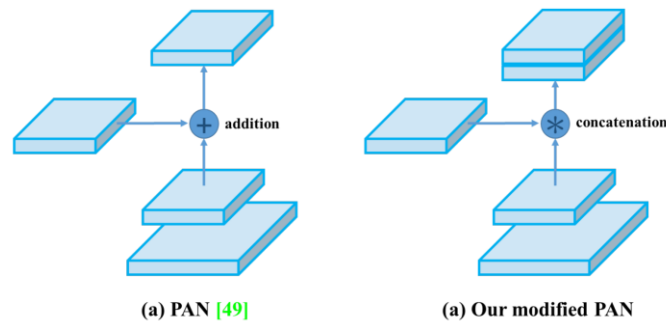


Рисунок 2.7 – Модифікована технологія PAN в YOLOv4.

Візуальна увага широко застосовується при виявленні об'єктів. *Spatial Attention Module (SAM)* зосереджується на тому, де знаходиться інформативна частина і відповідно формує просторову карту уваги. Однак, в YOLOv4 SAM теж модифікована і використовується без застосування максимізаційного й усереднювального агрегування.

2.1.3 Методи збільшення точності в базовій частині

До технологій, які покращують точність розпізнавання можна віднести 4 методи.

- CutMix Augmentation
- Mosaic Data Augmentation
- DropBlock Regularization
- Label Smoothing Regularization

CutMix використовується для прирощення даних. За аналогією з функцією Cutout, яка вирізає частину зображення для того щоби модель не була надмірно впевнена, CutMix не залишає пустих областей. У CutMix частина з одного зображення вирізається і вставляється в інше (рис.2.8) [18].





	ResNet-50	Mixup [48]	Cutout [3]	CutMix
Image				
Label	Dog 1.0	Dog 0.5 Cat 0.5	Dog 1.0	Dog 0.6 Cat 0.4
ImageNet Cls (%)	76.3 (+0.0)	77.4 (+1.1)	77.1 (+0.8)	78.6 (+2.3)
ImageNet Loc (%)	46.3 (+0.0)	45.8 (-0.5)	46.7 (+0.4)	47.3 (+1.0)
Pascal VOC Det (mAP)	75.6 (+0.0)	73.9 (-1.7)	75.1 (-0.5)	76.7 (+1.1)

Рисунок 2.8 – Порівняння результатів функцій прирощення даних.

Оскільки область замінюється на інше зображення із набору даних, кількість інформації та ефективність навчання не повинні сильно постраждати і точність може суттєво піднятися.

Ще один метод аугментації набору даних – *Mosaic*, який об'єднує чотири зображення в одне в форматі колажу.

У повнозв'язних шарах ми можемо застосувати виключення (dropout), щоби змусити модель навчатися на безлічі ознак замість того, щоби бути занадто впевненою в декількох. Однак це може не працювати для згорткових шарів. І до того ж, навіть якщо деякі пікселі виключені, просторову інформацію все ще можна виявити (рис.2.9 (b)).

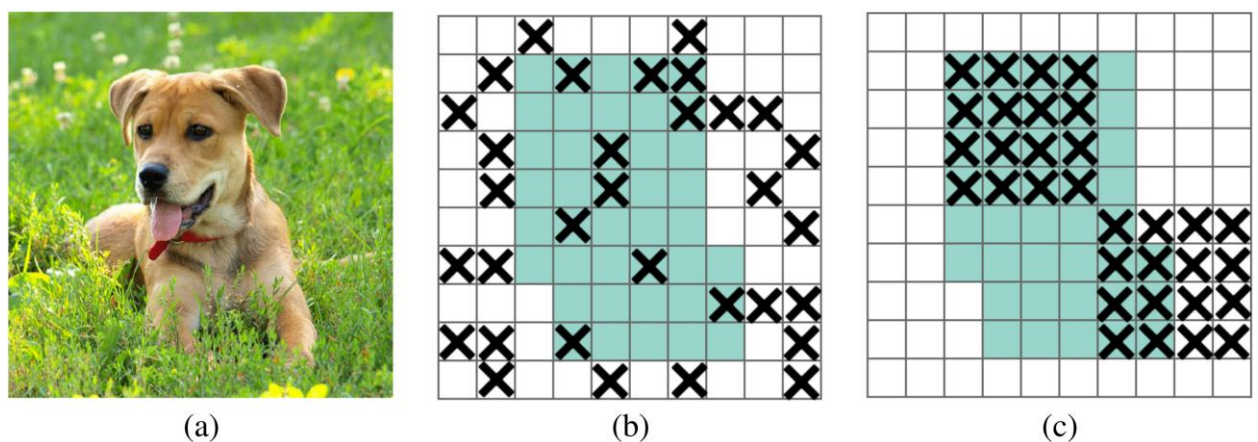


Рисунок 2.9 – Регуляризація DropBlock

Регуляризація *DropBlock* ґрунтується на аналогічній концепції, однак замість того, щоби відкидати окремі пікселі, відкидається блок пікселів (рис.2.9 (с)) [19].

Ще один підхід запобігти перенавчання мережі – техніка *Label Smoothing*. Вона знижує верхню межу прогнозу до нижчого значення, щоби при досягненні 100 % впевненості в прогнозі, модель не запам'ятовувала дані, а продовжувала навчатися. Таким чином, при розрахунку втрат буде використовуватися значення відмінне за 1,0.

2.1.4 Методи збільшення швидкості в базовій частині

Технології, які покращують продуктивність розпізнавання: Функції активації Mish та Swish.

За експериментальними даними, функція активації *Swish* (рис.2.10), показує кращу продуктивність, ніж ReLU і багато інших [20]. Передавальна функція Swish визначається як:

$$f(x) = x \cdot \sigma(\beta x),$$

де $\sigma(z) = (1 + \exp(-z))^{-1}$ – сигмоїдна функція;

β – або постійний, або навчений параметр.

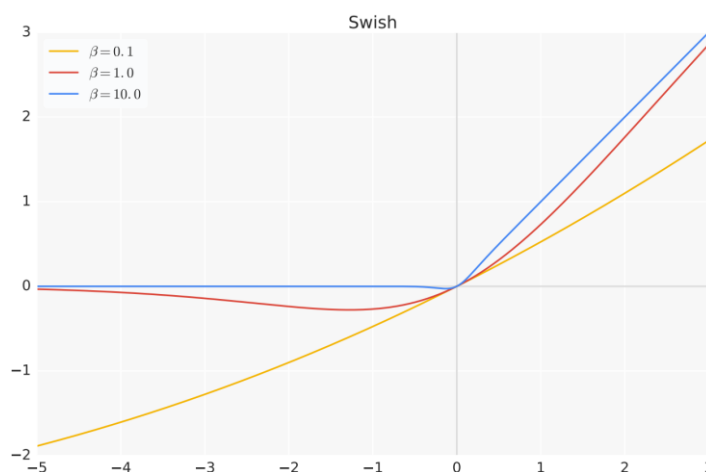


Рисунок 2.10 – Функція активації Swish

Mish – ще одна функція активації, дуже схожа на ReLU і Swish (Рис.2.11). Як стверджують автори, *Mish* може перевершувати їх у багатьох нейронних мережах і з різними наборами даних [21]. Передавальна функція *Mish* визначається як:

$$f(x) = x \cdot \tanh(\text{softplus}(x)),$$

де $\text{softplus}(x) = \ln(1 + e^x)$ – гладке наближення ReLU.

Swish, подібно *Mish* обмежена знизу і необмежена зверху з діапазоном $[\approx -0,31; +\infty)$.

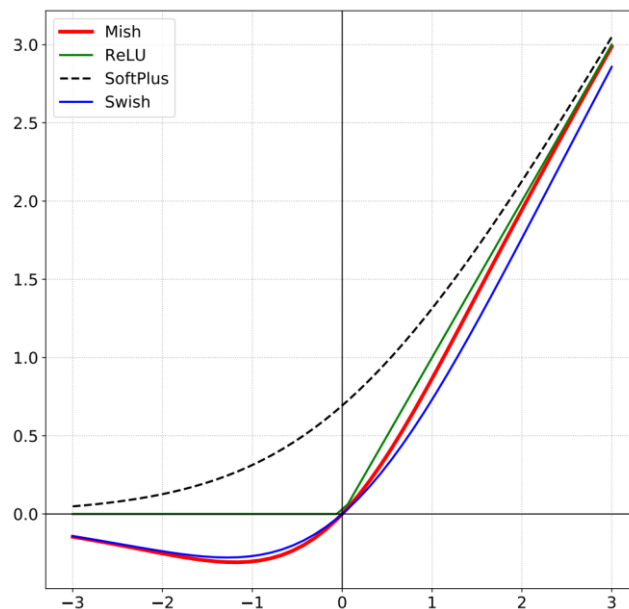


Рисунок 2.11 – Графік функцій активацій *Mish*, ReLU, SoftPlus та Swish.

2.1.5 Методи збільшення точності в головній частині

Технології, які використовуються в головній частині для покращення точності розпізнавання:

- Функція помилки CIoU
- Пакетна нормалізація CmBN
- Self-Adversarial Training
- Розподіл косинусного «відпалу»

- Еволюційні алгоритми вибору гіперпараметрів
- Random training shapes

Функції втрат підказують як коригувати ваги, щоби знизити витрати. Відповідно, коли відбуваються помилкові прогнози, функції вказують у якому напрямку рухатися. Однак в YOLO, для оцінки втрат використовується коефіцієнт Жаккару, або в галузі комп'ютерного зору – коефіцієнт перекриття вікон *IoU* (*Intersection over Union*) (2.1), і функція втрат не може з упевненістю передбачити напрямок. Наприклад, є кілька передбачуваних граничних рамок, які не перетинаються з істинною, і функція втрат не може визначити яка з рамок краще, хоча одна може бути ближче до еталонної.

$$IoU = \frac{|B \cap B^{gt}|}{|B \cup B^{gt}|}, \quad (2.1)$$

де $B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ – істинні (ground truth) граничні рамки;

$B = (x, y, w, h)$ – передбачувані граничні рамки.

Generalized IoU (*GIoU*) (2.2) виправляє цю проблему, удосконалюючи IoU. Функція, спочатку, розширює граничну рамку передбачення, поки вона не почне перекриватися з істинною. Потім вона стискається, щоби збільшити показник IoU. Цей процес вимагає більшої кількості ітерацій, часу й ресурсів.

$$\mathcal{L}_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|}, \quad (2.2)$$

де C – найменша рамка, що покриває B і B^{gt} .

Для досягнення більш швидкої збіжності між передбаченою рамкою і вихідною, використовується функція *Distance-IoU* (*DIoU*) (2.3).

$$\mathcal{L}_{DIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2}, \quad (2.3)$$

де b та b^{gt} – позначають центральні точки;

$\rho(\cdot)$ – евклідова відстань;

c – довжина діагоналі найменшої оточуючий рамки, що перекриває дві рамки.

І фінальна версія функції помилки *Complete IoU (CIoU)* (2.4), яка робить регресію більш точною, швидкою і поєднує в собі позитивні сторони попередніх функцій [22]:

- Збільшує площу перетину істинної рамки і передбачуваної рамки
- Мінімізує відстань до центральних точок рамок
- Підтримує постійне співвідношення меж рамок

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v, \quad (2.4)$$

де α – позитивний компромісний параметр;

$v = \frac{4}{\pi^2} \left(\arctan \frac{w^{gt}}{h^{gt}} - \arctan \frac{w}{h} \right)^2$ – вимірює узгодженість співвідношення рамок.

Звичайна пакетна нормалізація (batch normalization) збирає середнє значення (mean) і дисперсію (variances) зразків у мініпартії, щоби надати систематизацію для вхідного шару. Однак, якщо розмір мініпартії невеликий, ці оцінки будуть зашумленими. З кожною ітерацією ваги змінюються і статистика, зібрана на попередньому кроці, втрачається. На щастя, ці зміни ваг закономірні. *Cross-Iteration Batch Normalization (CBN)* розраховує статистику з k попередніх ітерацій із певним коригуванням (рис.2.12). А *CmBN* – це модифікована опція, яка збирає статистику тільки між мініпартіями в межах однієї партії [23].

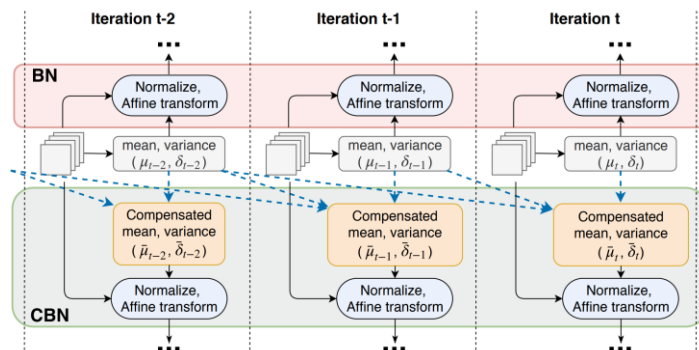
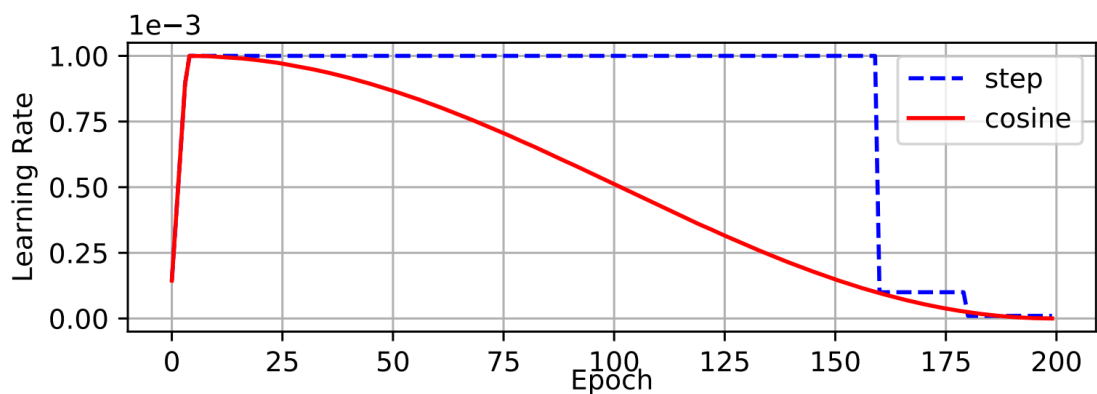


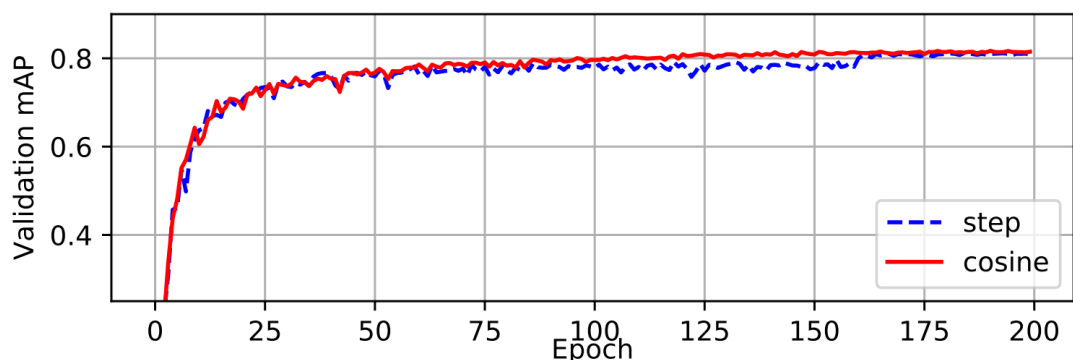
Рисунок 2.12 – Зображення пакетної нормалізації і запропонована CBN.

Традиційно, у методі зворотного поширення помилки коригують ваги моделі, щоби поліпшити виявлення об'єктів на зображенні. Однак, у методі збільшення обсягу даних *Self-Adversarial Training (SAT)* усе навпаки. Він змінює зображення таким чином (хоча візуальних змін можна не помітити), щоби погіршити роботу детектора. За допомогою цього нового зображення граничної рамки і класу, модель навчається. Це також надає невизначеності і зменшує перенавчання.

В YOLOv4 норма навчання відбувається відповідно *функції косинуса*. Спочатку велика швидкість навчання повільно починає знижуватися. Потім, всередині навчання швидкість збільшується і під кінець нахил функції вирівнюється, що призводить до низької швидкості навчання.



а) Функції норми навчання



б) Оцінка показника точності

Рисунок 2.13 – Візуалізація розподілу швидкості навчання ступінчастою і косинусною функцією.

На Рис.2.13 графічно зображено, як відбувається зниження швидкості навчання косинусної і ступінчастою функцією (а) і їхній вплив на показник точності mAP (б) [24].

Еволюційні алгоритми – це емпіричні методи засновані на припущенні, які працюють за принципом виживання найбільш пристосованих. Випадковим чином вибирається певна кількість наборів гіперпараметрів і на їхній основі навчаються моделі. Потім відбираються найкращі моделі й заново змінюються їхні гіперпараметри. Дані ітерацій тривають доти, поки не будуть знайдені найкращі параметри.

2.1.6 Методи збільшення швидкості в головній частині

У головній частині YOLOv4 використовуються раніше розглянуті властивості:

- Функція активації Mish
- Модифікований блок SPP
- Модифікований блок SAM
- Модифікований блок PAN

Фільтр придушення не максимумів відфільтровує граничні рамки, які прогнозують один і той же об'єкт і залишає одну з найвищою вірогідністю. Функція втрат DIoU (2.3) використовується в пошуку максимумів для відсіювання надлишкових обмежувальних рамок. Він враховує показник IoU і відстань між центральними точками двох рамок. Це дає більш точний результат пошуку при оклюзіях.

2.2 Архітектура і конфігурація YOLOv5

YOLOv5 була представлена 25 червня 2020 року, іншою компанією – Ultralytics, через кілька місяців після релізу YOLOv4. У даному підрозділі розглянемо залучені технології в її архітектуру.

Архітектура YOLOv5 складається з тих же трьох основних каркасів, які застосовувалися в попередніх версіях YOLO:

- Базова частина (backbone)
- Шийна частина (neck)
- Головна частина (head)

Найбільший внесок YOLOv5 – це зміна програмного середовища з Darknet на PyTorch. Darknet в основному написаний на C і пропонує більш тонкий контроль над операціями, закладених у нейромережу. У багатьох випадках використання низькорівневої мови є перевагою для дослідження та збору інформації, однак, з іншого боку, це може уповільнити втілення нових ідей.

До того ж PyTorch дає змогу вдвічі знизити точність обчислень із плаваючою комою під час навчання й логічному виводу з 32-бітної до 16-бітної точності. Це значно прискорює час обробки. Однак, на даний момент, збільшення швидкості є тільки на серверних графічних процесорах, а саме на V100 і T4. Проте, NVIDIA заявила про намір збільшити покриття пристроїв, які будуть підтримувати таку можливість.

У попередніх версіях рамки прив'язки необов'язково, але бажано було ставити самостійно, що могло викликати труднощі й певні похибки при помилковому розрахунку. У YOLOv5 розробники представили концепцію автоматичних рамок прив'язки *AutoAnchor*. При завантаженні даних, *AutoAnchor*, або використовує вже задані рамки, або підбирає стандартні й починає порівнювати їх з істинним до граничного значення. Якщо вони не підходять, вони продовжують налаштовуватися, використовуючи методи k-середніх і алгоритмів генетичного навчання. Врешті, алгоритм підставляє знайдені рамки в набір даних і продовжує навчання.

Як і в YOLOv4, YOLOv5 використовується базова частина, що заснована на архітектурі CSPNet. CSP усуває проблему поширення градієнта

в інші частини згорткової мережі, що призводить до меншої кількості витрачених ресурсів і можливості робити обчислення на менш продуктивних пристроях. Це надзвичайно важливо для сімейства YOLO, де швидкість виведення та невеликий розмір моделі мають первинне значення.

Для приращення даних в YOLOv5 використовується кілька видів змін зображень: масштабування, коригування колірного простору і аугментація мозаїкою.

В іншому, архітектура YOLOv5 залишається такою ж як і в YOLOv4 включаючи шийну частину, в якій використовується модифікований метод PAN.

2.3 Порівняльний аналіз версій YOLOv4 та YOLOv5

Перші версії YOLO випускалися впродовж трьох років. Однак лише за кілька місяців у 2020 році були представлені 3 різні версії від різних розробників: YOLOv4, YOLOv5 і PP-YOLO. Випуск YOLOv5 навіть спричинив суперечки серед експертів у сфері машинного навчання та комп'ютерного зору. У даному підрозділі порівнюються версії: Darknet YOLOv4 і PyTorch YOLOv5, їхні поліпшення, швидкість та багато інших характеристик, про які буде сказано далі.

Щоб оцінити якість, кожна нова архітектура детекторів об'єктів оцінюється на їхню продуктивність. У науковій сфері прийнято порівнювати показник точності (mAP) моделей на одному наборі даних MS COCO з однаковими зображеннями. Однак, під час експлуатації в реальному світі, показник точності не єдиний параметр першорядної важливості. У кожній предметної галузі, показники можуть варіюватися залежно від поставлених проблем.

При порівнянні моделей навчання відбуватиметься на наборі даних компанії FLIR, і враховуватися наскільки довго проходить навчання,

наскільки вони точні, наскільки швидкий вивід, наскільки складно налаштувати модель для виробничо середовища, і т. ін.. І так, порівняння буде проходити за наступними критеріями:

- Порівняння гнучкості налаштування конфігурацій навчання
- Порівняння часу навчання
- Порівняння точності
- Порівняння розмірів навчених моделей
- Порівняння часу обробки

Навчання всіх моделей YOLO відбувалося в Google Colaboratory (Colab) – інструмент для редагування й виконання Python коду в браузері, який відмінно підходить для задач машинного навчання. Colab забезпечує вільний доступ до обчислювальних ресурсів, включаючи графічний прискорювач NVIDIA Tesla T4.

Набір даних для навчання

В оцінюванні використовувався набір даних компанії FLIR для виявлення транспортних засобів. Набір складається з близько 8000 інфрачервоних зображень, які збиралися з тепловізійних камер, встановлених на автономному транспорті [25]. В таблиці 2.1 наведені більш обширні характеристики.

Таблиця 2.1 – Характеристика набору даних FLIR.

<i>Загальна кількість образів</i>	Понад 14 тисяч зображень Понад 10 тисяч коротких відео Понад 4 тисячі 140-секундних відео
<i>Частота оновлення зображення</i>	Записано в 30 Гц. Відеоанотації виконувались із записом 30 кадрів/сек.
<i>Зображення з граничними рамками</i>	9 214 кадрів з обмежувальними рамками. 1. Люди (28 151)

Продовження Табл.2.1

<i>Зображення з граничними рамками</i>	2. Автомобілі (46 692) 3. Велосипеди (4 457) 4. Собаки (240) 5. Інші транспортні засоби (2228)
<i>Відео з граничними рамками</i>	4. Собаки (0) 5. Інші транспортні засоби (540)
<i>Відео з граничними рамками</i>	4,183 кадрів з обмежувальними рамками. 1. Люди (21,965) 2. Автомобілі (14,013) 3. Велосипеди (1,205)

Порівняння гнучкості налаштування конфігурацій навчання

У YOLOv4 і YOLOv5 використовуються конфігураційні файли (з розширенням .cfg і .yaml), які визначають архітектуру і параметри навчання. У конфігурації Darknet вказується безліч додаткових відомостей про навчання, починаючи від кількості класів (classes), ітерацій навчання (steps) й розміру партії (batch) і закінчуючи детальним налагодженням кожного шару нейронної мережі, зміна функцій активації, технологій аугментації даних(cutmix, mosaic, gaussian_noise), та ін.[26, 27]. З цими параметрами можна експериментувати, щоби досягти кращих результатів у різних ситуаціях.

Конфігурація YOLOv5 менш гнучка. Є можливість змінювати стандартне налаштування кількості класів, а також трохи налагоджувати архітектуру шарів головної і основної частини нейронної мережі.

Файли конфігурації двох версій YOLO дають змогу досить гнучко вносити зміни на архітектурному рівні, проте Darknet версія забезпечує більшу свободу дій і тому, є найкращим вибором для дослідницьких цілей.

Порівняння часу навчання

В YOLOv4 тривалість навчання багато в чому залежить від кількості ітерацій (*max_batches*), який залежить від кількості класів. Розробники рекомендують використовувати вираз:

$$max_batches = classes * 2000 = 4 * 2000 = 8000$$

отже, кількість ітерацій для YOLOv4 – 8000.

YOLOv5 використовує епохи для визначення тривалості навчання. Спираючись на YOLOv4 визначимо кількість епох:

$$epoch = \frac{max_batches * batch * accumulate}{images} = \frac{8000 * 32 * 2}{8862} \approx 58$$

Приймемо наступні загальні параметри навчання для всіх моделей YOLO:

- *batch* = 32 – кількість зображень в одному пакеті,
- *width, height* = 416x416 – розмір мережі,
- *classes* = 4 – кількість заявлених класів в наборі
- *subdivisions* = 8 – кількість зображень у мініпакеті.

З цими параметрами YOLOv4 навчається близько 400 хвилин на наборі FLIR. Однак, максимальне значення точності досягнуто на 3200 ітерації, на виконання яких, витрачено 162 хвилини. Решта моделей навчалися рівно задану кількість ітерацій / епох і ці результати наведені на рис.2.14.

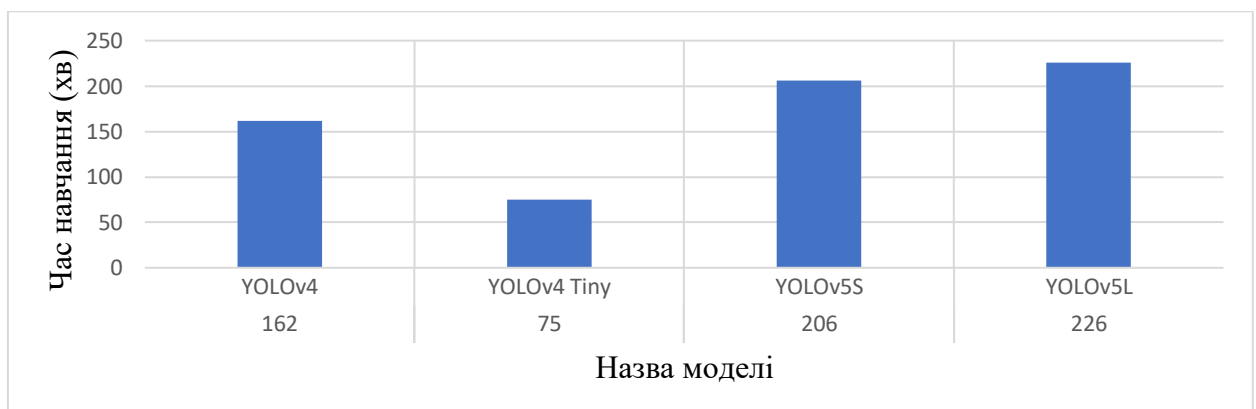


Рисунок 2.14 – Порівняння часу навчання моделей YOLO

Порівняння точності

У задачі порівняння точності використовується параметр $mAP@0.5$, який оцінюється знаходженням області під кривою влучності (precision) – з усіх виявлених рамок, яка частина була дійсно коректна, і повнотою (recall) – з усіх коректно виявлених рамок, яка частина була знайдена з усього набору.

Гістограма порівняння випробовуваних версій YOLO приведена на рис.2.15. Значення AP для кожного класу, а також інші метрики оцінки Darknet версій YOLOv4 та Tiny наведені на рис.2.16 (а, б). І відповідно значення для PyTorch версії на рис.2.16 (в-д).

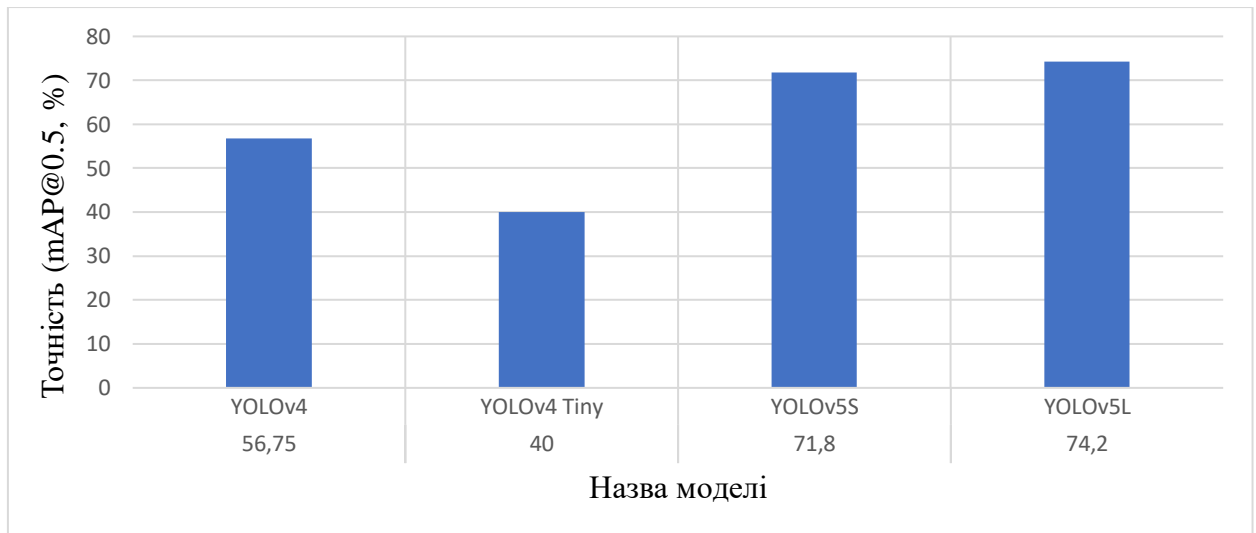


Рисунок 2.15 – Порівняння точності моделей YOLO

```
class_id = 0, name = person, ap = 70.46%      (TP = 4053, FP = 2117)
class_id = 1, name = bicycle, ap = 65.12%    (TP = 309, FP = 155)
class_id = 2, name = car, ap = 87.28%        (TP = 4605, FP = 1500)
class_id = 3, name = dog, ap = 4.16%         (TP = 2, FP = 12)

for conf_thresh = 0.25, precision = 0.70, recall = 0.77, F1-score = 0.73
for conf_thresh = 0.25, TP = 8969, FP = 3784, FN = 2727, average IoU = 52.09 %

IoU threshold = 50 %, used Area-Under-Curve for each unique Recall
mean average precision (mAP@0.50) = 0.567546, or 56.75 %
```

(a) – YOLOv4

```

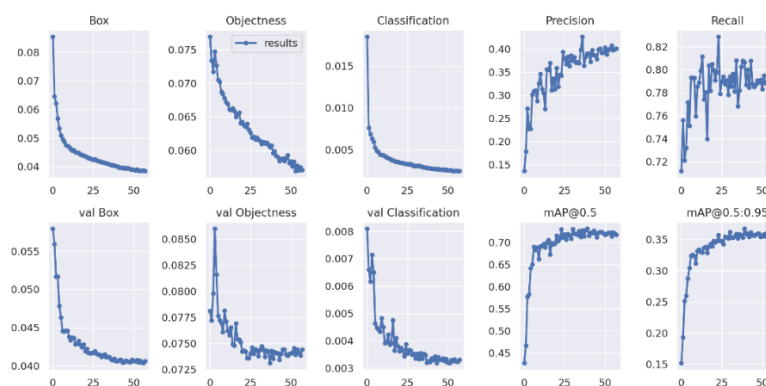
class_id = 0, name = person, ap = 46.00%      (TP = 3089, FP = 2672)
class_id = 1, name = bicycle, ap = 37.14%    (TP = 224, FP = 284)
class_id = 2, name = car, ap = 77.15%        (TP = 4090, FP = 1766)
class_id = 3, name = dog, ap = 0.04%         (TP = 0, FP = 0)

for conf_thresh = 0.25, precision = 0.61, recall = 0.63, F1-score = 0.62
for conf_thresh = 0.25, TP = 7403, FP = 4722, FN = 4293, average IoU = 44.70 %

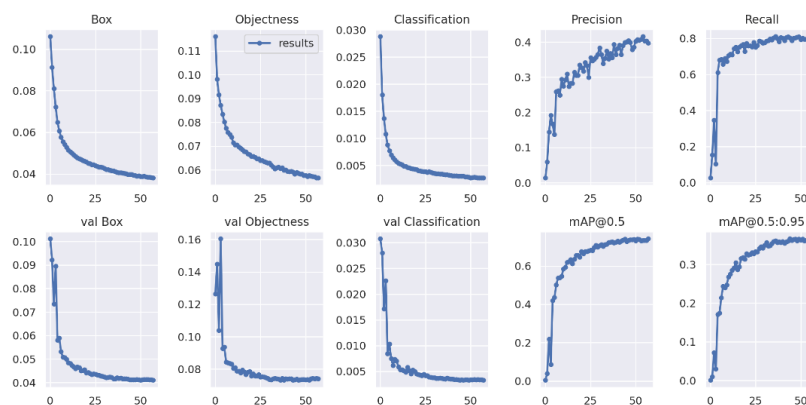
IoU threshold = 50 %, used Area-Under-Curve for each unique Recall
mean average precision (mAP@0.50) = 0.400837, or 40.08 %

```

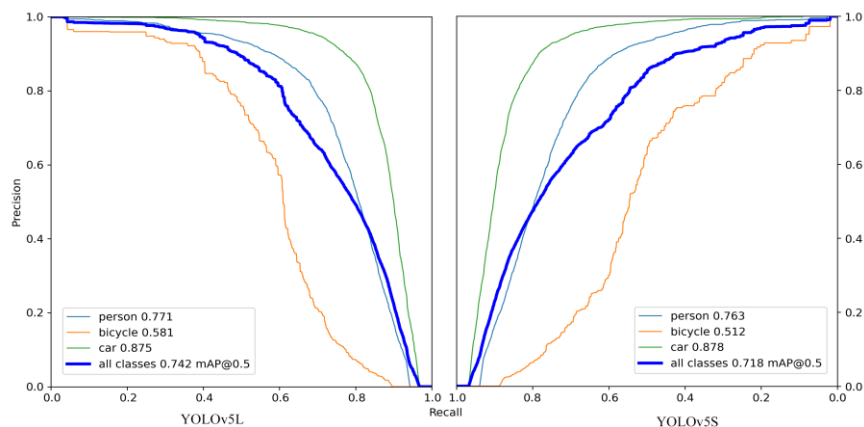
(б) – YOLOv4Tiny



(в) – YOLOv5S



(г) – YOLOv5L



(д)

Рисунок 2.16 – Додаткові метрики оцінки точності.

Результати точності YOLOv5 сильно випереджають Darknet версії, однак це не свідчить про перевагу PyTorch версії. Ймовірно, YOLOv5 справляється краще в конкретно поставленому завданні, навченою на основі певного набору даних. Але, слід зауважити, що на графіку влучності до повноти (Рис.2.16 (д)), клас собак навіть не враховується, отже показник середньої точності, ймовірно, хибний. Якщо подивитися на порівняння моделей YOLO (Рис.2.17) на оціночному наборі MS COCO з великою кількістю класів, YOLOv4 перевершує п'яту версію. Однак варто зазначити, що в цьому порівнянні, YOLOv4 навчалася в середовищі TensorFlow (не у власному Darknet) в форматі числа половинної точності.

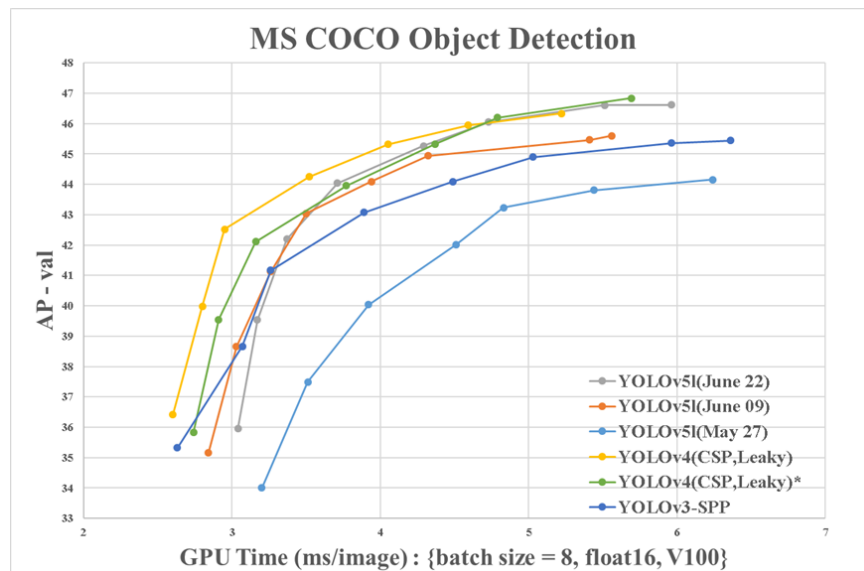


Рисунок 2.17 – Порівняння версій YOLO на наборі даних MS COCO.

Порівняння розмірів навчених моделей

Розмір вагових коефіцієнтів – досить важливий параметр для перенесення моделі на портативні та вбудовані пристрої. Через те, що версії YOLOv4tiny і YOLOv5s якраз побудовані для подібних задач, вони мають мінімальний розмір файлу. Порівняння з великими моделями наведено на Рис.2.18.

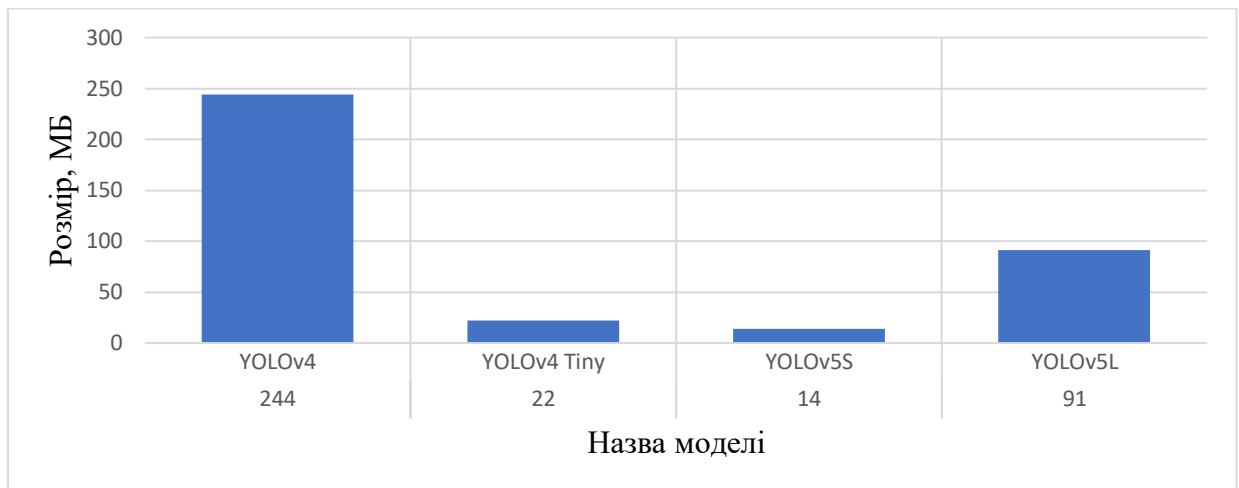


Рисунок 2.18 – Порівняння розмірів моделей YOLO

Порівняння часу обробки

Для визначення швидкості роботи нейронних мереж будуть використані пакети з одним зображенням (batch = 1) і розміром 512x512. Це дозволить досягти максимально наближених результатів із реальними промисловими системами. Швидкість роботи можна збільшити за допомогою спеціалізованого програмного забезпечення (TensorRT), яке вимагає особливі процесори з тензорними ядрами, однак не у всіх системах присутні подібні рішення. Час обробки тестувався на серверному й десктопному прискорювачі NVIDIA Tesla T4 і NVIDIA RTX2060S, відповідно. Результати порівняння наведені на Рис.2.19.

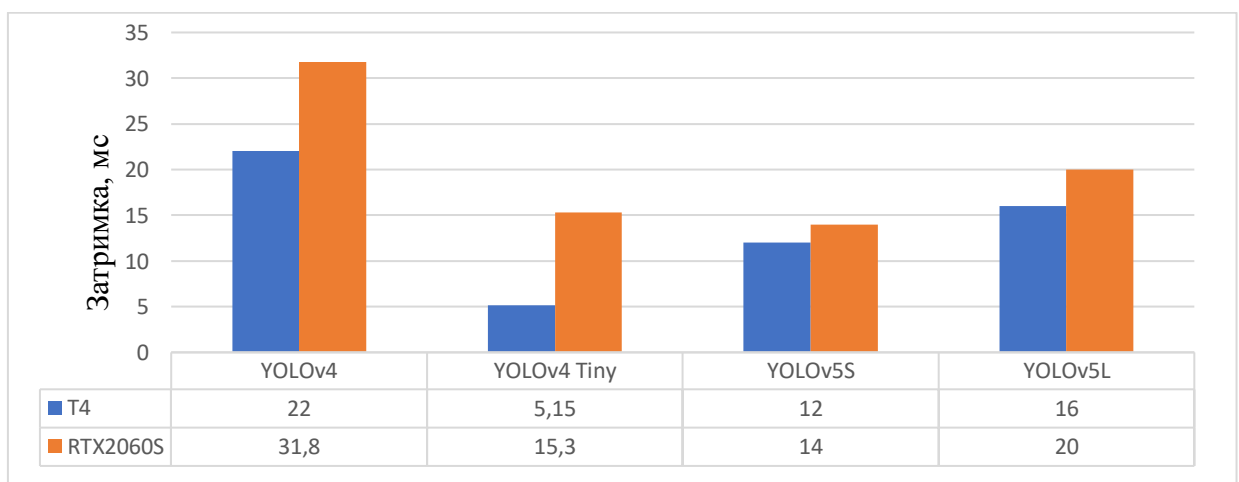


Рисунок 2.19 – Порівняння часу обробки моделей YOLO

Висновки за розд. 2

1. В даному розділі докладно розглянута архітектура YOLOv4, які новітні технології були змінені і інтегровані. Досліджено технології, що поліпшують процеси навчання в кожній частині мережі, а саме точність і швидкість. До таких технологій належать:

- CutMix, Mosaic – аугментації зображень.
- DropBlock, Label Smoothing, Self-Adversarial Training – запобігання від перенавчання.
- DIOU, GIoU, CIOU – функції втрат.
- Mish, Swish – функції активації.
- Darknet-53, CSP, SAM, SPP – архітектури об'єднання шарів нейромережі.
- CmBN – пакетна нормалізація.
- Косинусна норма навчання.
- Еволюційні алгоритми вибору гіперпараметрів.

2. Розглянуто PyTorch реалізацію YOLO (YOLOv5), і які переваги в порівнянні з Darknet версією вона надає. Переведення на інший фреймворк дозволяє збільшити швидкість роботи за рахунок зниженої точності обчислень. Оглянуто суттєве нововведення YOLOv5 – AutoAnchor, яке на кожній ітерації автоматично підлаштовує рамки прив'язки в залежності від набору даних, що неможливо в YOLOv4.

3. Проведено детальний аналіз чотирьох передових версій YOLO на даний момент: повноцінні багат шарові версії YOLOv4 і YOLOv5L; молодші і спрощені версії – YOLOv4Tiny, YOLOv5S. Аналіз проводився в однакових умовах, на серверному графічному прискорювачі Tesla T4 і десктопному RTX2060S за критеріями точності, швидкості, розміром вагових коефіцієнтів, часу навчання і гнучкості налаштування.

Найбільше часу на навчання витратила модель YOLOv4 (162 хв.) найменше – YOLOv4 Tiny (75 хв). PyTorch версії показали приблизно схожі результати, незважаючи на різні складності архітектури (~216 хв).

Розмір моделі YOLOv4 перевищує інші в кілька разів, молодші версії не сильно відрізняються один від одного в своїй категорії (не перевищують 22 МБ), старша версія YOLOv5L – 91 МБ.

Аналіз точності mAP@0.5 показав, що PyTorch версії перевершують Darknet (56,75% та 40% проти 74,2% та 71,8%), однак графіки відношення влучності до повноти побудовані без урахування одного класу. Можливо, даних про цей класі було настільки мало, що програмне середовище порахувала його помилково заданим і не включало його до опрацювання.

В результатах швидкості обробки зображень не виникло нетипових показань. Затримка обробки на серверному прискорювачі менше, ніж на десктопному. Darknet YOLO показала відмінний результат молодшої версії (5,15 мс і 15,3 мс), але старша має найбільшу затримку з усіх (22 мс і 31,8 мс). YOLOv5 показала середні результати (12 – 20) мс.

За результатами аналізу можна стверджувати, що всі моделі прийнятно впоралися із завданням і в кожній категорії мають свої переваги та недоліки. Однак, версія Darknet YOLO є більш гнучкою до налаштувань і відкриває широкі можливості для покращення на власному наборі даних. Тому для використання в дисертаційних дослідженнях обрано четверту версію YOLO на програмному середовищі Darknet.

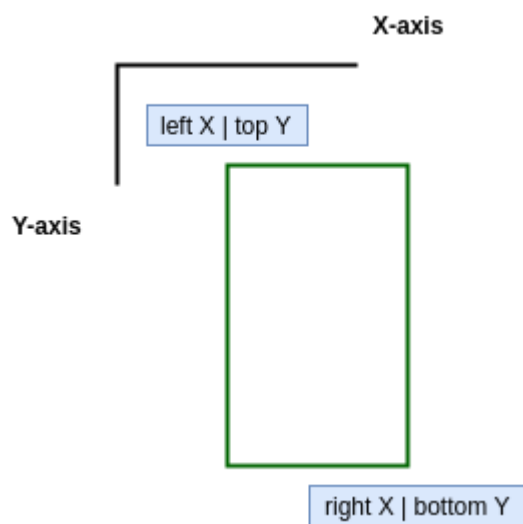
РОЗДІЛ 3. МЕТОД ТА АЛГОРИТМИ ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ НАВЧАННЯ МОДЕЛЕЙ YOLO НА ПЕРСОНАЛЬНОМУ НАБОРІ ДАНИХ

За результатами другого розділу для дослідження розпізнавання обрана YOLOv4 в програмній сфері Darknet. Характеристики цього детектора недостатньо великі, щоби використовувати в реальних умовах. Нижче запропоновані підходи, які збільшили показники молодшої версії YOLOv4Tiny та YOLOv4. Також, проведена порівняльна характеристика, що показує помітний приріст ефективності.

3.1 Оптимізація і конвертація анотацій зображень

У цьому дослідженні довелося провести кілька попередніх змін над форматом анотацій зображень, щоб отримати бажаний результат. У вихідному наборі даних YOLO очікує відносних значень щодо ширини та висоти зображення, тобто координати обмежувальних рамок задаються наступним чином (рис.3.1):

X_{Min} , X_{Max} , Y_{Min} , Y_{Max} – координати вікна в нормалізованих координатах зображення. X_{Min} знаходиться в діапазоні $[0,1]$, де 0 – крайній лівий піксель, а 1 – крайній правий піксель на зображенні. Координати Y рухаються від верхнього пікселя (0) до нижнього пікселя (1).



Однак, у початковому наборі даних координати граничних рамок відповідають фактичній кількості пікселів відповідного зображення. Крім цього, кожна анотація має назву класу об'єкта в текстовому форматі, у той час як YOLO приймає тільки числові значення. Отже, створений скрипт

Рисунок 3.1 – Координати обмежувальних рамок

(Рис.3.2) нормалізує значення координат і конвертує ім'я класу в числовий формат.

Алгоритм конвертації координат

```

1: for для всіх зображень  $i$  do
2:   for для анотацій  $ann$  do
3:     if зображення в  $ann = i$  and категорія в  $ann \leq 3$  then
4:        $cat\_id \leftarrow$  номер категорії
5:        $left, right, bbox\_w, bbox\_h \leftarrow$  значення граничної рамки
6:       /* Значення класів YOLO починається з нульового індексу */
7:        $cat\_id \leftarrow cat\_id - 1$ 
8:       /* Розрахунок центрів ширини та висоти граничної рамки */
9:        $x\_center \leftarrow left + bbox\_w/2$ 
10:       $y\_center \leftarrow right + bbox\_h/2$ 
11:      /* Розрахунок значень відносно розміру зображення */
12:       $x\_rel, y\_rel \leftarrow (x\_center/width, y\_center/height)$ 
13:       $w\_rel, h\_rel \leftarrow (bbox\_w/width, bbox\_h/height)$ 
14:      /* Збереження отриманого результату */
15:       $Result \leftarrow (cat\_id, x\_rel, y\_rel, w\_rel, h\_rel)$ 
16:     end if
17:   end for
18: end for

```

Рисунок 3.2 – Частина лістингу алгоритма

3.2 Додаткова аугментація зображень

Набір даних FLIR від самого початку незбалансований. У специфікації вказано, що існує п'ять категорій (люди, велосипеди, автомобілі, собаки і великі вантажівки), але тільки три категорії (люди, велосипеди й автомобілі) можуть бути досить помітні за допомогою анотацій. Розподіл класів у вихідних наборах

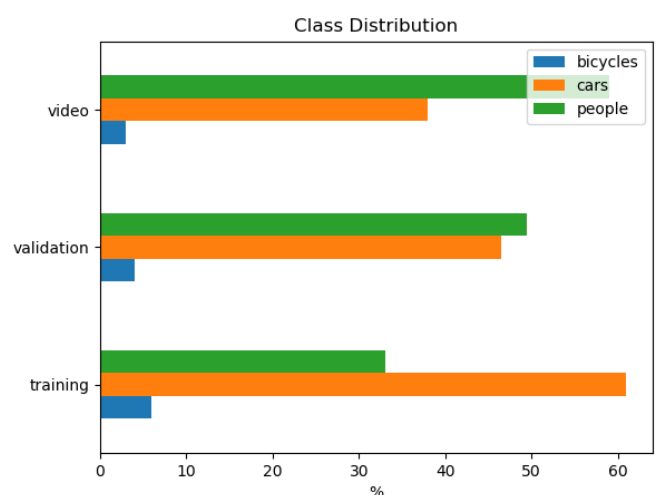


Рисунок 3.3 – Розподілення класів в наборі даних FLIR

даних (відео, перевірка й навчання) виглядає наступним чином (Рис.3.3)

Один зі способів змусити мережу більше розкривати рідкісний клас – кілька разів помістити одні й ті ж кадри, що включають цей клас, у навчальний набір. Завдяки цим заходам вдалося досягти збільшення точності класу велосипедів.

3.3 Роль рамок прив'язки в YOLO та їх налаштування

YOLO, як і інші одноетапні методи виявлення об'єктів використовують рамки прив'язки. Це зумовлені обмежувальні рамки, форма й розмір, яких відповідає об'єктами в наборі навчальних даних. Вони генеруються за допомогою кластеризації еталонних граничних рамок із вихідного набору даних, щоби знайти більш поширені форми й розміри. Розробники використовували кластеризацію k-середніх. Це зумовлено швидкістю й ефективністю в порівнянні з іншими алгоритмами. Також, через створення великої кількості кластерів, результат виходить точнішим.

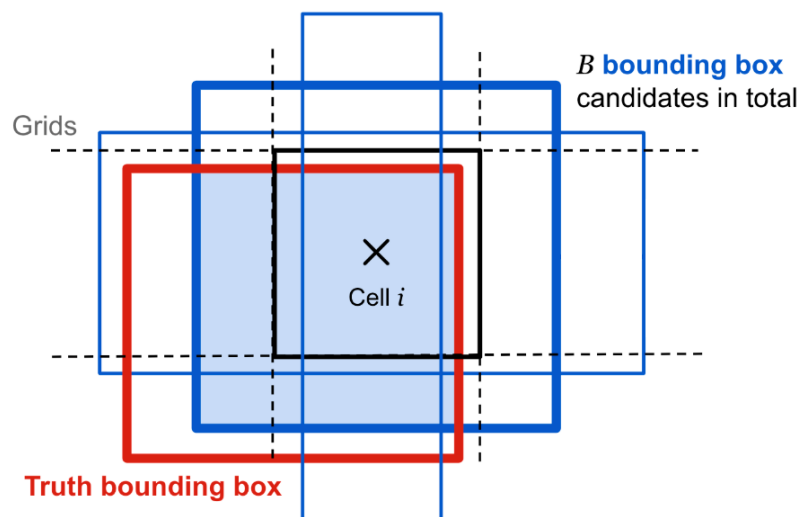


Рисунок 3.4 – Наглядне розміщення рамок прив'язки

Як уже згадувалося раніше, зображення розбивається на сітку різної розмірності. До кожного осередку присвоюються різні рамки прив'язки з однаковим центром (Рис.3.4). І в подальшому навчанні, замість випадкового

передбачення координат, буде розраховуватися зміщення граничних рамок. Через те, що рамки прив'язки побудовані на основі дійсних, пошук координат зводиться до мінімуму.

3.4 Налаштування конфігурації нейронної мережі

Важливим кроком при навчанні нейронної мережі є коректна настройка конфігураційного файлу. У цьому файлі містяться всі основні параметри архітектури і треба налаштовувати його з особливою обережністю, оскільки помилки можуть спричиняти серйозні проблеми з навчанням і навіть розтягнути процес на кілька днів.

Розглянемо основні параметри конфігураційного файлу.

Batch = 64 – кількість зразків (зображень), які будуть оброблені в одній партії.

Subdivisions = 16 – кількість *mini batches* в одній партії *batch*, розмір *mini batch* = *batch* / *subdivisions*. GPU обробляє зразки *mini batch* одночасно, а ваги будуть оновлюватися одразу для всіх пакетних зразків (1 ітерація обробляє серію зображень).

Width x Height = 512 x 512 – розмір мережі (ширина x висота), кожне зображення буде змінено до розміру мережі під час навчання та виявлення.

Classes = 3 – кількість класів об'єктів, які можуть бути виявлені. Для нашого випадку: люди, автомобілі, велосипеди.

Max batches = 6000 – задає загальну кількість усіх ітерацій для навчання.

$$\text{Max batches} = \text{Classes} * 2000$$

Steps = 4800, 5400 – при цій кількості ітерацій, швидкість навчання буде помножена на коефіцієнт масштабування.

$$Steps = (80\% \text{ of } max_batches), (90\% \text{ of } max_batches)$$

$Filters = 24$ – кількість фільтрів ядра, які можна розрахувати за наступною формулою:

$$Filters = (Classes + 5) * B,$$

YOLOv4 передбачає три обмежувальні рамки для кожного осередку карти функцій, отже, значення B дорівнює 3, і формула для нашого випадку має такий вигляд:

$$Filters = (3 + 5) * 3 = 24$$

Окрім конфігураційного файлу, прийнято рішення навчити мережу на розширеному наборі даних із використанням попередньо навчених ваг згортки із репозиторія розробника. Попередньо треновані ваги дають дві переваги:

- Мережі потрібно менше часу для сходження.
- Краща оцінка mAP навіть при першому підрахунку.

3.5 Моделювання та оцінка результатів навчання моделі нейронної мережі підвищеної ефективності

Для оцінювання точності нейронної мережі будемо поступово підключати функції та параметри алгоритму, щоби визначити, як це впливає на результати. Отже, можна знайти ключову відмінність, що приносить найбільші поліпшення. Для тренування також використовувався сервіс Colab із графічним прискорювачем Tesla T4. Перший результат тренування без додаткових налаштувань наведено на Рис.3.5.

У документації FLIR наведено, що базова точність даних навчання й перевірки була встановлена за допомогою нейронної мережі RefineDetect512, розробленої для зображень 512x512 і попередньо навченої на даних MS COCO. Значення mAP були отримані для категорій Люди (0,794),

Велосипеди (0,580) і Машини (0,856). Отриманий неоптимізований результат уже перевершує показники точності.

```
class_id = 0, name = person, ap = 83.71%      (TP = 4786, FP = 1766)
class_id = 1, name = bicycle, ap = 66.31%     (TP = 314, FP = 174)
class_id = 2, name = car, ap = 89.50%         (TP = 4820, FP = 1728)

for conf_thresh = 0.25, precision = 0.73, recall = 0.85, F1-score = 0.79
for conf_thresh = 0.25, TP = 9920, FP = 3668, FN = 1762, average IoU = 57.05 %

IoU threshold = 50 %, used Area-Under-Curve for each unique Recall
mean average precision (mAP@0.50) = 0.798393, or 79.84 %
Total Detection Time: 43 Seconds
```

Рисунок 3.5– Результат тренування зі стандартною конфігурацією

Наступним кроком додамо власні значення точок прив'язок і розширимо навчальний набір зображеннями, які включають клас велосипедів.

Таблиця 3.1 – Порівняння точності

Клас	RD 512	Без налаштувань		Точки прив'язки		Фінальний результат	
		YOLOv4	YOLOv4 Tiny	YOLOv4	YOLOv4 Tiny	YOLOv4	YOLOv4 Tiny
Люди	0,794	0,837	0,571	0,842	0,650	0,863	0,649
Велосипеди	0,580	0,663	0,410	0,695	0,479	0,826	0,514
Машини	0,856	0,895	0,789	0,893	0,813	0,912	0,820
mAP@0.5	0,743	0,798	0,590	0,810	0,648	0,875	0,661

У табл. 3.1 наведені показники точності для всіх ітерацій і моделей YOLO. Можна помітити, як змінився фінальний результат після дублювання зображень з однаковим класом. Крім збільшення точності виявлення велосипедів, зросла й точність виявлення інших класів. Більш змістовні результати останнього навчання наведені нижче на Рис.3.6.

```

class_id = 0, name = person, ap = 86.29%      (TP = 4819, FP = 1721)
class_id = 1, name = bicycle, ap = 82.64%    (TP = 3412, FP = 1138)
class_id = 2, name = car, ap = 91.16%        (TP = 4965, FP = 1638)

for conf_thresh = 0.25, precision = 0.80, recall = 0.87, F1-score = 0.84
for conf_thresh = 0.25, TP = 13196, FP = 4497, FN = 3159, average IoU = 64.28 %

IoU threshold = 50 %, used Area-Under-Curve for each unique Recall
mean average precision (mAP@0.50) = 0.874764, or 87.48 %
Total Detection Time: 38 Seconds

```

Рис.3.6 – Фінальний результат тренування включаючи всі параметри

Визначення затримки виведення проводилося за умовами 2 розділу. Час обробки порівнювався на десктопному й серверному графічних прискорювачах і з розміром batch = 1 в середовищі Darknet без використання TensorRT + tkDNN (Табл. 3.2).

Таблиця 3.2 – Затримка виведення зображень

Розмір	Tesla T4		RTX 2060S	
	YOLOv4	YOLOv4Tiny	YOLOv4	YOLOv4Tiny
320x320	17,2 мс	2,3 мс	15,58 мс	9,09 мс
416x416	21,3 мс	5,12 мс	30,6 мс	11,5 мс
512x512	25,1 мс	7,5 мс	31 мс	12,55 мс
640x512	24,2 мс	8,8 мс	31,25 мс	14,24 мс

Наведемо зображення (Рис.3.7 – Рис.3.9), які оброблені нейронною мережею YOLOv4 (зліва), і зображення з істинними граничними рамками з набору даних FLIR (справа).

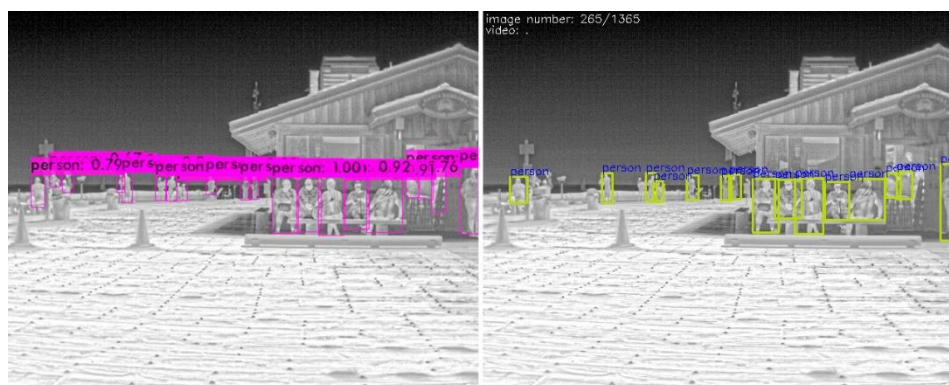


Рисунок 3.7 – Порівняння результатів

На рисунках можна побачити, що невеликі об'єкти (з класу Люди) не були анотовані в наборі даних FLIR, але виявлялися мережею. Це може збільшувати кількість помилкових спрацьовувань (FP), особливо в класі людей, що призводить до більш низького балу mAP. Звичайно, також спостерігається кілька хибно-негативних (FN) прикладів на невеликих об'єктах, але $FP > FN$ за браком анотацій. І це теж сприяє втратам і впливає на градієнти.



Рисунок 3.8 – Порівняння результатів



Рисунок 3.9 – Порівняння результатів

Висновки за розд. 3

У даному розділі проаналізовані різні варіанти реалізації виявлення теплових об'єктів за допомогою нейронної мережі YOLOv4. Анотації зображень оптимізовані й конвертовані в необхідний формат.

Прийнято кілька рішень, які збільшили показник точності mAP:

1. Відеодані включені в навчальний набір.
2. Збережені тільки три категорії, оскільки FLIR оцінює модель тільки на основі трьох категорій (люди, велосипеди і автомобілі).
3. Зображення із класом велосипедів були продубльовані і включені в навчальний набір, що значно поліпшило mAP для категорії велосипедів (66,31% \rightarrow 82,64%).
4. Навчання мережі відбувалося з використанням попередньо навчених вагових коефіцієнтів згортки.

Фінальні числові результати точності та швидкості на прискорювачах Tesla T4/RTX2060S для YOLOv4: mAP@0,5 = 87,5%, FPS = 41/32 кадрів/с; для YOLOv4Tiny: mAP@0,5 = 66,1%, FPS = 198/70 кадрів/с. Показники цілком задовільні, враховуючи вихідний бал точності FLIR (74,3%) і швидкість обробки більша за 30 кадрів/с.

РОЗДІЛ 4. РОЗРОБЛЕННЯ СТАРТАП-ПРОЄКТУ

4.1. Сутність ідеї стартапу та технологічного рішення

Кількість реалізацій систем розпізнавання об'єктів із кожним роком тільки збільшується. Існує безліч реалізацій, які застосовуються у сферах безпеки в громадських місцях і будівлях та в системах автономного водіння. Найчастіше зустрічаються класифікація об'єктів, розпізнавання облич і жестів. Розпізнавання облич ідентифікує й підтверджує особистість людини. Воно аналізує обличчя людини й порівнює шаблони на основі закріплених деталей. Розпізнавання жестів працює за схожим принципом, але, як впливає з назви, аналізуються не обличчя, а рухи людини. Це може бути корисним у спортивних змаганнях, розвагах і людних місцях, наприклад, щоби виявити агресивну поведінку або критичні ситуації в натовпі.

Однак, систем працюють здебільшого лише у видимому світловому діапазоні, відповідно в нічний час доби, схожі інструменти можуть відчувати труднощі з детектуванням. Запропоноване рішення вирішує проблему відсутності світла, бо здатне виявляти об'єкти в інфрачервоному діапазоні з тепловізійних камер. У табл. 4.1 вказані можливі напрямки використання проєкту.

Таблиця 4.1 – Напрямки використання стартап-проєкту

Апаратні засоби	Сфери застосування	Напрямки застосування
Оптико-електронні пошуково-оглядові комплекси, що працюють в ІЧ діапазоні світла	1. Безпілотні літальні апарати 2. Автономні транспортні засоби. 3. Системи безпеки та спостереження.	1. Пошук та виявлення об'єктів. 2. Відстеження об'єктів 3. Підрахунок об'єктів

Проект розроблявся з акцентом на продуктивність і можливість застосування на портативних пристроях. Технологічні параметри наведені в табл. 4.2.

Таблиця 4.2 – Технологічні рішення

Технологічна ідея	Реалізація рішення	Наявність технології
Гарна швидкість розпізнавання для свого класу	Використання спрощеної архітектури нейронної мережі.	Наявна
Підвищена ефективність розпізнавання об'єктів	Використання великого набору даних і унікальна настройка параметрів для навчання нейронної мережі.	Наявна

Висновки: головною ідеєю стартап проекту є – високоефективна система розпізнавання об'єктів в ІЧ діапазоні, яка може бути застосоване, як у державній сфері, так і в споживчій.

4.2. Техніко-економічна характеристика продукції

Наведемо унікальні особливості стартап проекту, які можуть бути фактором конкурентної переваги (Табл. 4.3). До технічних характеристик можна віднести параметри продуктивності, ефективності, точності та інші. Для заданого випадку, ефективність – це середній час обробки 1-го зображення. До економічних відносяться собівартість, вартість підтримки та обслуговування. Показники W, N, S ілюструють значущість параметра. W – слабка значущість, N – нейтральна значущість, S – сильна значущість.

Таблиця 4.3 - Визначення сильних, слабких та нейтральних характеристик

№ n/n	Техніко- економічні характеристики ідеї	(потенційні) товари/концепції конкурентів				W	N	S
		Проект	Конку- рент 1	Конку- рент 2	Конку- рент 3			
1.	Швидкість	≥ 31 кадрів/с	≤ 30 кадрів/с	≤ 25 кадрів/с	≤ 25 кадрів/с		+	
2.	Точність	87,5%	70%	66,31%	82,9%			+
4.	ІЧ діапазон	є	нема	є	нема			+
5.	Собівартість	10	20	30	25		+	
6.	Підтримка	є	є	є	є	+		
7.	Обслуговування	є	є	є	є	+		

Висновки: порівнюючи конкурентні реалізації, параметр точності перевершує розроблювану модель нейронної мережі, але ефективність роботи набагато краще, що впливає на економічних показниках підтримки та обслуговування.

4.3. Актуальність проєкту та аналіз ринку

Таблиця 4.4 – Попередня характеристика потенційного ринку стартап-проєкту

№ n/n	Показники стану ринку (найменування)	Характеристика
1.	Кількість головних гравців, од	3
2.	Загальний обсяг продаж, грн/ум.од	7500
3.	Динаміка ринку	Зростає
4.	Наявність обмежень для входу	Немає

Продовження Табл. 4.4

<i>№ n/n</i>	<i>Показники стану ринку (найменування)</i>	<i>Характеристика</i>
5.	Специфічні вимоги до стандартизації та сертифікації	Відсутні
6.	Середня норма рентабельності в галузі (або по ринку), %	65%

Висновки: за попередніми характеристикам можна зробити висновки, що ринок у галузі виявлення об'єктів зростає й конкурентів поки що недостатньо, щоби виникли серйозні проблеми з запуском торговельної компанії.

Таблиця 4.5 - Характеристика потенційних клієнтів стартап-проєкту

<i>№ n/n</i>	<i>Потреба, що формує ринок</i>	<i>Цільова аудиторія (цільові сегменти ринку)</i>	<i>Відмінності у поведінці різних потенційних цільових груп клієнтів</i>	<i>Вимоги споживачів до товару</i>
1.	Виявлення об'єктів в ІЧ діапазоні.	Державний/ приватний сектор	Інтеграція з камерами нічного бачення, оптичними системи в транспортних засобах, тепловізорами та інш.	Ефективність, точність, тривалість роботи, легкість використання.

Висновки: реалізацію якісної моделі нейронної мережі для розпізнавання об'єктів, яку можна використовувати в портативних пристроях може залучити, як державний сектор, так і приватні підприємства, які прагнуть інтегрувати систему для додаткового захисту та безпеки.

Таблиця 4.6 - Фактори загрози

<i>№ n/n</i>	<i>Фактор</i>	<i>Зміст загрози</i>	<i>Можлива реакція компанії</i>
1.	Економічний	Економічний стан країни виробника	Зміна країни виробника
2.	Якісний	Не належна якість розроблювальної системи	Вдосконалення процесів виявлення
3.	Конкуренція	Найбільш відомі гравці, що закріпилися на ринку	Проведення маркетингової кампанії.
4.	Політичний	Політична ситуація країни виробника	Зміна країни виробника

Висновки: Хоча конкурентів у цій галузі поки не багато, значну роль загрози виходу на ринок представляють компанії, які закріпилися в цій області протягом декількох років і буде важко схилити клієнтів у наш бік.

Таблиця 4.7 - Фактори можливостей

<i>№ n/n</i>	<i>Фактор</i>	<i>Зміст можливості</i>	<i>Можлива реакція компанії</i>
1.	Поширення	Впровадження технології в системах відеоспостереження, автомобілях та інших галузях.	Співпраця з компаніями з відповідних галузей.
2.	Розширення функціоналу	Можливість додавання нових класів об'єктів до існуючих. Визначення ворожих / небезпечних дій, аналізування моделей руху для поліпшення передбачення поведінки	Розширення штату співробітників та додавання окремих підрозділів.

Продовження Табл. 4.7

<i>№ n/n</i>	<i>Фактор</i>	<i>Зміст можливості</i>	<i>Можлива реакція компанії</i>
3.	Індивідуальні замовлення	Можливість виконувати індивідуальні потреби замовників	Аналіз замовлень
4.	Злиття із лідерами ринку	Можливість злиття компанії із конкурентами	Аналіз та оцінка можливих позитивних і негативних факторів.

Висновки: У сфері комп'ютерного зору існує безліч можливостей, які допоможуть досягти, або злиття компанії з іншими гравцями на ринку, або економічної переваги над конкурентами.

Таблиця 4.8 - Ступеневий аналіз конкуренції на ринку

<i>Особливості конкурентного середовища</i>	<i>В чому проявляється дана характеристика</i>	<i>Вплив на діяльність підприємства</i>
Тип конкуренції – олігополія	Врахування реакції конкурентів.	Постійно стежити за зміною ринку.
За рівнем конкурентної боротьби – міжнародна.	Замовники з декількох країн.	Вихід на міжнародний ринок
За галузевою ознакою – міжгалузева	Система застосовується в різних галузях.	Прагнення захопити більше галузей.
Конкуренція за видами товарів – товарно-ротова	Конкуренція різноманітних моделей детекторів об'єктів.	Орієнтація стратегії компанії на клієнта та адаптація до змін ринкових умов

Продовження Табл. 4.8

<i>Особливості конкурентного середовища</i>	<i>В чому проявляється дана характеристика</i>	<i>Вплив на діяльність підприємства</i>
За характеристиками конкурентних переваг – нецінова	Залучення нових споживачів ефективністю роботи.	Вчасне вдосконалення продукту
За інтенсивністю – товарна	Конкуренція між функціями товару	Вдосконалення функцій продукту.

Висновки: проаналізувавши пропонування, визначили загальні риси конкуренції на ринку нейромережових детекторів об'єктів. Через зростаючий ринок, з'являється все більше нових міжнародних конкурентів, які можуть створювати свої моделі під конкретні завдання, тим самим задаючи свої умови і формуючи олігопольний тип конкуренції.

Таблиця 4.9 - Аналіз конкуренції в галузі за М. Портером

	<i>Прямі конкуренти в галузі</i>	<i>Потенційні конкуренти</i>	<i>Клієнти</i>	<i>Товари замітники</i>
<i>Складові аналізу</i>	AnyVision CloudWalk Clarifai SCYLLA	FLIR Dahua Technology	Державний та приватний сектори	Компанії, що спеціалізуються на виявленні об'єктів у видимому діапазоні світла
<i>Висновки</i>	Існують прямі конкуренти, які спеціалізуються на виявленні об'єктів в ІЧ діапазоні.	Конкуренти, які займаються тільки тепловізійним обладнанням і потенційно небезпечні	Умови ринку другорядно залежать від клієнтів	Компанії можуть почати розробляти системи виявлення в ІЧ діапазоні

Таблиця 4.10 - Обґрунтування факторів конкурентоспроможності

<i>№ n/ n</i>	<i>Фактор конкурентоспроможності</i>	<i>Обґрунтування</i>
1.	Собівартість	Швидше поширення продукту.
2.	Ефективність	Ефективність дозволяє охопити більшу кількість пристроїв.
3.	Точність	Вища точність в своєму класі привертає більше клієнтів.

Висновки: з огляду на аналіз конкуренції, вимоги клієнтів і ідеї проекту визначили основні конкурентні фактори. Собівартість дасть нагоду поліпшувати продукт і відповідно знаходити більше замовників. Ефективність і точність збільшує якість роботи системи, що позитивно впливає на ліквідність товару.

Таблиця 4.11 - Порівняльний аналіз сильних та слабких сторін проекту

<i>№ n/n</i>	<i>Фактор конкурентоспроможності</i>	<i>Бали 1-20</i>	<i>Рейтинг розроблювальної системи у порівнянні з товарами конкурентів</i>						
			-3	-2	-1	0	+1	+2	+3
1.	Собівартість	12		-2					
2.	Ефективність	19	-3						
3.	Точність	15			-1				

Висновки: За рейтингом сильних та слабких сторін проекту, розроблювальне рішення перевершує конкурентні за собівартістю, ефективністю та точністю.

Таблиця 4.12 - SWOT-аналіз стартап-проекту

Сильні сторона: Ефективність Багатоплатформеність	Слабкі сторони: Обмеженість ІЧ діапазоном світла Низька репутація на ринку
---	--

Продовження Табл.4.12

Можливості: Збільшення споживчої бази та попиту на товар.	Загрози: Конкуренція Труднощі з поширенням продукту
--	---

Висновки: SWOT-аналіз показав сильні і слабкі сторони проєкту і які можливості, або загрози можуть виникнути, якщо їх враховувати.

Таблиця 4.13 - Альтернативи ринкового впровадження стартап-проєкту

<i>№ n/n</i>	<i>Альтернатива ринкової поведінки</i>	<i>Ймовірність отримання ресурсів</i>	<i>Строки реалізації</i>
1.	Рекламна кампанія	Висока	6 місяців
2.	Участь в міжнародних конференціях	Середня	6 місяців

Висновки: З огляду на проведений SWOT-аналіз, обрані альтернативні заходи впровадження стартап-проєкту.

4.4. Розроблення ринкової та маркетингової стратегії проєкту

Таблиця 4.14 - Вибір цільових груп потенційних споживачів

<i>№ n/n</i>	<i>Опис профілю цільової групи потенційних клієнтів</i>	<i>Готовність споживачів сприйняти продукт</i>	<i>Орієнтовний попит в межах цільової групи (сегменту)</i>	<i>Інтенсивність конкуренції в сегменті</i>	<i>Простота входу у сегмент</i>
1.	Системи безпеки та спостереження	+	+	Середня	+

Продовження Табл.4.14

2.	Транспортні засоби.	+	+	Середня	+
----	---------------------	---	---	---------	---

Висновок: Галузі безпеки та спостереження, а також галузь транспортних засобів обрані в якості цільових груп потенційних споживачів, бо ці сегменти готові сприймати продукт та присутня здорова конкуренція.

Таблиця 4.15 - Визначення базової стратегії розвитку

<i>№ п/п</i>	<i>Обрана альтернатива розвитку проєкту</i>	<i>Стратегія охоплення ринку</i>	<i>Ключові конкурентоспроможні позиції до обраної альтернативи</i>	<i>Базова стратегія розвитку</i>
1.	Рекламна кампанія	Стратегія диференціації	Виробництво більш якісного продукту	Стратегія спеціалізації

Висновок: Базова стратегія розвитку – спеціалізація, яка орієнтована не на весь ринок, а тільки на один цільовий сегмент. Альтернативна стратегія – диференціації, передбачає більше витрат і відповідно більше рентабельності за рахунок ринку, який зможе прийняти високу ціну.

Таблиця 4.16 — Визначення базової стратегії конкурентної поведінки

<i>№ п/п</i>	<i>Чи є проєкт «першопрохідцем» на ринку?</i>	<i>Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?</i>	<i>Чи буде компанія копіювати основні характеристики товару конкурента, і які?</i>	<i>Стратегія конку- рентної поведінки</i>
1.	Не є «першопрохідцем»	Буде шукати нових споживачів, і забирати існуючих	Може використовуватися схожа архітектура нейронної мережі.	Стратегія наслідування лідера

Висновок: Проєкт не є першопрохідцем та використовує схожу архітектуру нейронної мережі, але гнучкість застосування дозволяє вибрати стратегію наслідування лідера.

Таблиця 4.17 - Визначення стратегії позиціонування

<i>№ n/n</i>	<i>Вимоги до товару цільової аудиторії</i>	<i>Базова стратегія розвитку</i>	<i>Ключові конкуренто- спроможні позиції власного стартап- проєкту</i>	<i>Вибір асоціацій, які мають сформувати комплексну позицію власного проєкту (три ключових)</i>
1.	Ефектив- ність	Стратегія спеціалізації	Швидкість розпізнавання	Висока швидкодія роботи
2.	Точність	Стратегія спеціалізації	Якість	Висока точність роботи

Висновки: Стратегія позиціонування ґрунтується на ефективності та точності, які формують ключові позиції стартап-проєкту – швидкість та якість роботи.

Таблиця 4.18 - Визначення ключових переваг концепції потенційного товару

<i>№ n/n</i>	<i>Потреба</i>	<i>Вигода, яку пропонує товар</i>	<i>Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)</i>
1.	Ефективність	Висока швидкість роботи на будь-якому пристрої	Ціна, точність, ефективність, легкість використання.
2.	Точність	Висока точність.	

Висновок: Проаналізувавши результати конкурентоспроможності товару, можна почати формувати маркетингову концепцію.

Таблиця 4.19 - Опис трьох рівнів моделі товару

<i>Рівні товару</i>	<i>Сутність та складові</i>		
Товар за задумом	Система розпізнавання об'єктів на ІЧ зображеннях і відео в реальному часі з використанням сучасних нейронних мереж.		
Товар у реальному виконанні	Властивості/характеристики	М/Нм	Вр/Тх/Тл/Е/Ор
	Ефективність	М	Тх
	Точність	М	Тх
	Якість: відповідає нормам ISO/IEC 25010:2014 «Інженерія систем та програмного забезпечення - Вимоги до якості систем та програмного забезпечення»		
Товар із підкріпленням	Пакування: стиснення файлу програмного забезпечення		
	До продажу – програмне забезпечення, яке встановлюється на кінцевий пристрій користувача.		
	Після продажу – сервісна допомога.		

Висновок: Сформувавши маркетингову модель товару, визначимо метод його захисту від копіювання. Програмне забезпечення буде поширюватися за стратегією ліцензування заснованої на функціональності, яка дозволяє відкривати додаткові можливості за певну плату.

Таблиця 4.20 - Визначення меж встановлення ціни

<i>№ п/ п</i>	<i>Рівень цін на товари замітники</i>	<i>Рівень цін на товари-аналоги</i>	<i>Рівень доходів цільової групи споживачів</i>	<i>Верхня та нижня межі встановлення ціни на товар/послугу</i>
1.	300-2000 у.о.	0-500 у.о./місяць	500-5000 у.о.	0-300 у.о.

Висновок: Аналізуючи цінові межі подібних послуг на ринку, можна зробити висновок, що існує великий розкид від безкоштовних, але з обмеженим функціоналом, або багатофункціональні корпоративні версії з відповідною ціною. Система буде містити технічні засоби захисту авторських прав (DRM).

Таблиця 4.21 - Формування системи збуту

<i>№ п/ п</i>	<i>Специфіка закупівельної поведінки цільових клієнтів</i>	<i>Функції збуту, які має виконувати постачальник товару</i>	<i>Глибина каналу збуту</i>	<i>Оптимальна система збуту</i>
1.	Безкоштовний доступ до програми і відкриття додаткових можливостей за окрему плату.	Зворотній зв'язок із замовниками	Нульового/ першого рівня	Прямий, через сайт виробника
2.	Продаж фізичних копій програми	Постачання товару до кінцевого замовника	Нульового/ першого рівня	Прямий, через сайт виробника

Висновок: Товар може бути доставлений замовникам двома способами. Перший – засобами цифрової дистрибуції через магазини додатків, або безпосередньо через сайт розробника. Другий – продаж програмного забезпечення на матеріальних носіях, через сайт розробника.

Таблиця 4.22 - Концепція маркетингових комунікацій

<i>№ п/п</i>	<i>Специфіка поведінки цільових клієнтів</i>	<i>Канали комунікацій, якими користу- ються цільові клієнти</i>	<i>Ключові позиції, обрані для позиціону- вання</i>	<i>Завдання рекламного повідом- лення</i>	<i>Концепція рекламного звернення</i>
1.	Бажання поліпшити системи безпеки	Соціальні мережі, ЗМІ, партнерство, e-mail, рекомендації	Ефектив- ність роботи, якість розпізна- вання, доступність на більшості пристроях, легкість користу- вання	Розповсюдж- ення інформації про продукт.	Опис принципу роботи системи у науково- професій- ному стилі, відео демонст- рація налашту- вання та роботи
2.	Отримання додаткових чинників для вдоскона- лення розпізна- вання.				

Висновок: Основним каналом комунікації для потенційних клієнтів є соціальні мережі, на які буде направлено більше зусиль і коштів.

Висновки за розд. 4

За результатами розділу розробки стартап-проєкту можна зробити такі висновки:

1. Розроблений метод може бути застосований у пошуково-оглядових ОЕС для відстеження й розпізнавання об'єктів. Наведені потенційно-можливі класифікації ринку за територіальною ознакою (глобальний) і суб'єктам, які потенційно можуть бути замовниками (ринок споживачів, виробників і державних установ).

2. Наведені технічні й економічні характеристики, які визначають переваги розроблюваного рішення від конкурентних. Ринок розпізнавання об'єктів активно зростає й конкуренти, хоча і присутні, але не мають спричиняти серйозних проблем із запуском торговельної компанії. Завдяки активній сфері комп'ютерного зору наявні широкі можливості розширення або злиття стартап-проєкту з іншими гравцями.

3. Розроблена початкова маркетингова стратегія. Сформовано стратегії розвитку, позиціонування, а також описані три рівні моделі, встановлені цінові межі і способи збуту.

Загальні висновки

У дисертаційній роботі вирішено актуальну та важливу науково-прикладну задачу підвищення ефективності (за критеріями точності та швидкодії) розпізнавання об'єктів на інфрачервоних зображеннях та відеопослідовностях шляхом подальшого розвитку методу розпізнавання об'єктів на основі згорткових нейромереж типу YOLO та алгоритмічних і програмних рішень його реалізації. Під час виконання досліджень отримано наступні практичні та наукові результати:

1. Проведено аналіз існуючих проблем виявлення об'єктів в оптико-електронних тепловізійних пристроях, який показав, що найбільший негативний внесок у розпізнавання вносять візуальні перешкоди, нерівномірний фон, погодні умови й людський фактор спостерігача.

2. Досліджено відомі патентні технічні рішення, що використовують процедуру розпізнавання об'єктів в інфрачервоному діапазоні та встановлено відсутність у більшості пристроїв швидкісного завадостійкого каналу передачі відеоданих, що важливо при розміщенні джерела відеоінформації на певній відстані (до 15 м) від обчислювального блоку чи блоку виведення пристрою.

3. Проведено аналіз сучасних одноетапних методів виявлення об'єктів, призначених для роботи в реальному часі та обґрунтовано вибір за критеріями швидкості і точності методу на основі архітектури YOLO в якості базового для дослідження.

4. Детально розглянуто різновиди сучасних архітектур YOLOv4 і YOLOv5, які використовують різні комплекси програмних рішень та шляхом їх порівняльного аналізу обґрунтовано вибір четвертої версії завдяки наявності в ній гнучкості налаштувань.

5. Удосконалено метод виявлення об'єктів на основі нейромережі YOLOv4 шляхом попередньої нормалізації і конвертації анотацій зображень,

дублювання зображень з однаковим класом та навчання мережі з використанням попередньо навчених вагових коефіцієнтів згортки, що дозволило підвищити ефективність (за критеріями точності та швидкодії) розпізнавання об'єктів між класами на інфрачервоних зображеннях та відеопослідовностях.

6. Розроблено алгоритмічні та програмні засоби реалізації методу підвищення ефективності розпізнавання згорткових мереж на базі архітектури YOLO і проведено вивчення та порівняльне моделювання нейронних мереж із застосуванням запропонованих рішень, яке підтвердило підвищення ефективності. Зокрема, точність розпізнавання YOLOv4 підвищилась з 79,8% до 87,5%. Затримка обробки одного зображення зменшилася з 31,8 мс (~31 кадрів/с) до 31 мс (~32 кадрів/с). Точність розпізнавання YOLOv4tiny підвищилась з 59% до 66,1%. Затримка обробки одного зображення зменшилася з 15,3 мс (~65 кадрів/с) до 14,24 мс (~70 кадрів/с).

7. Розроблений початковий етап стартап-проєкту дозволяє переконатися в можливості впровадження інноваційного рішення ефективного розпізнавання об'єктів на інфрачервоних зображеннях у виробничий процес.

Вищевикладене дозволяє зробити висновок, що завдання на магістерську дисертацію виконано у повному обсязі, а її мета досягнута.

За матеріалами досліджень опубліковано 1 друковану статтю (див. Додаток А):

– А. С. Лисенко. Аналіз факторів, які впливають на ефективність оптико-електронних систем // Електронна та Акустична Інженерія. – 2020. – Т. 3, №1. - С. 15 – 18.

Список використаних джерел

1. See the full spectrum. *Infiniti Electro-Optics* : веб-сайт. URL: <https://www.infinitioptics.com/technology/multi-sensor> (дата звернення: 15.12.2020)
2. See Heat. No Light Required. *Infiniti Electro-Optics* : веб-сайт. URL: <https://www.infinitioptics.com/technology/multi-sensor> (дата звернення: 15.12.2020)
3. Dragana P., Branko L., Miroslav P., Sasa V. Thermal Imager Range: Predictions, Expectations, and Reality. 2019. DOI: [10.3390/S19153313](https://doi.org/10.3390/S19153313)
4. Liu W., Anguelov D., Erhan D., Szegedy C., Reed S., Fu Ch.-Y., Berg A. C. SSD: Single Shot MultiBox Detector. Computer Vision – ECCV 2016, Springer. 2016. Vol. 9905. P. 21-37. DOI: [10.1007/978-3-319-46448-0_2](https://doi.org/10.1007/978-3-319-46448-0_2).
5. How single-shot detector (SSD) works? *ArcGIS* : веб-сайт. URL: <https://developers.arcgis.com/python/guide/how-ssd-works> (дата звернення: 15.12.2020)
6. Lin T., Dollár P., Girshick R., He K., Hariharan B., Belongie, S. Feature Pyramid Networks for Object Detection. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017. P. 936-944. DOI: 10.1109/CVPR.2017.106.
7. Ren S., He K., Girshick R., Sun J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2017. Vol. 39, №. 6, P. 1137-1149. DOI: 10.1109/TPAMI.2016.2577031.
8. He K., Gkioxari G., Dollar P., Girshick R. Mask R-CNN. 2017 IEEE International Conference on Computer Vision (ICCV). 2017. P. 2980-2988. DOI: 10.1109/ICCV.2017.322
9. Real E., Aggarwal A., Huang Y., Le Q. V. Regularized Evolution for Image Classifier Architecture Search. AAAI. 2019 Vol. 33. №. 01. P. 4780-4789.

DOI: 10.1609/aaai.v33i01.33014780

10. Lin T., Goyal P., Girshick R., He K., Dollár P. Focal Loss for Dense Object Detection. 2017 IEEE International Conference on Computer Vision (ICCV). 2017. P. 2999-3007. DOI: 10.1109/ICCV.2017.324.
11. Redmon J., Divvala S., Girshick R., Farhadi A. You Only Look Once: Unified, Real-Time Object Detection. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016. P. 779-788. DOI: 10.1109/CVPR.2016.91.
12. Redmon J., Farhadi A. YOLOv3: An Incremental Improvement. 2018. URL: <https://arxiv.org/abs/1804.02767v1> (дата звернення: 15.12.2020).
13. COCO Dataset. *COCO* : веб-сайт. URL: <https://cocodataset.org/#download> (дата звернення: 15.12.2020).
14. Bochkovski A., Wang Y., Liao H.-Y. M. YOLOv4: Optimal Speed and Accuracy of Object Detection. 2018. URL: <https://arxiv.org/abs/2004.10934> (дата звернення: 15.12.2020).
15. Huang G., Liu Z., Van Der Maaten L., Weinberger Q. Densely Connected Convolutional Networks. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017. P. 2261-2269. DOI: 10.1109/CVPR.2017.243.
16. Wang C., Mark Liao H., Wu Y., Chen P., Hsieh J, Yeh I. CSPNet: A New Backbone that can Enhance Learning Capability of CNN. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). 2020. P. 1571-1580. DOI: 10.1109/CVPRW50498.2020.00203.
17. Huang Z., Wang J., Fu X., Yu T., Guo Y., Wang R. DC-SPP-YOLO: Dense connection and spatial pyramid pooling based YOLO for object detection. Information Sciences. 2020. Vol. 522. P. 241-258. DOI: 10.1016/j.ins.2020.02.067
18. Yun S., Han D., Chun S., Oh S. J., Yoo Y., Choe J. CutMix: Regularization Strategy to Train Strong Classifiers With Localizable Features. 2019

- IEEE/CVF International Conference on Computer Vision (ICCV). 2019. P. 6022-6031. DOI: 10.1109/ICCV.2019.00612.
19. Ghiasi G., Lin T.-Y., Le Q. V. DropBlock: A regularization method for convolutional networks. 2018. URL: <https://arxiv.org/abs/1810.12890> (дата звернення: 15.12.2020).
 20. Ramachandran P., Zoph B., Le Q. V. Searching for Activation Functions. 2017. URL: <https://arxiv.org/abs/1710.05941> (дата звернення: 15.12.2020).
 21. Misra D. Mish: A Self Regularized Non-Monotonic Activation Function. 2019. URL: <https://arxiv.org/abs/1908.08681> (дата звернення: 15.12.2020).
 22. Zheng Z., Wang P., Liu W., Li J., Ye R., Ren D. Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression. AAAI 2020. 2020. Vol. 34. №. 07. P. 12993-13000.
 23. Yao Z., Cao Y., Zheng S., Huang G., Lin S. Cross-Iteration Batch Normalization. URL: <https://arxiv.org/abs/2002.05712> (дата звернення: 15.12.2020).
 24. Zhang Z., He T., Zhang H., Zhang Z., Xie J., Li M. Bag of Freebies for Training Object Detection Neural Networks. 2019. URL: <https://arxiv.org/abs/2002.05712> (дата звернення: 15.12.2020).
 25. FREE FLIR Thermal Dataset for Algorithm Training. *FLIR* : веб-сайт. URL: <https://www.flir.com/oem/adas/adas-dataset-form/> (дата звернення: 15.12.2020).
 26. Bochkovskiy A. CFG Parameters in the different layers. 2020. веб-сайт. URL: <https://github.com/AlexeyAB/darknet/wiki/CFG-Parameters-in-the-different-layers> (дата звернення: 15.12.2020).
 27. Bochkovskiy A. CFG Parameters in the [net] section. 2020. веб-сайт. URL: <https://github.com/AlexeyAB/darknet/wiki/CFG-Parameters-in-the-%5Bnet%5D-section> (дата звернення: 15.12.2020).

Аналіз факторів, які впливають на ефективність оптико-електронних систем

Лисенко А.С.

Кафедра конструювання електронно-обчислювальної апаратури keoa.kpi.ua

Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського» kpi.ua

Київ, Україна

Анотація—У даній статті розглянуто основні критерії виявлення об'єктів з використанням оптико-електронних засобів в інфрачервоному діапазоні. Проведено аналіз літературних джерел і наведено основи візуалізації і формування інфрачервоних зображень в тепловізійних системах. Розглянуто доцільність використання критерію Джонсона при розрахунку ефективності оптико-електронних систем в реальних умовах. Зазначені фактори, які в більшій мірі впливають на дальність і якість виявлення цілі. Також проаналізовані актуальні моделі та програмні пакети, які враховують негативні фактори.

Ключові слова — система спостереження; тепловізійна система; інфрачервоне зображення; виявлення об'єктів.

I. ВСТУП

На сьогодні оптико-електронні системи (ОЕС) з захопленням зображення в інфрачервоному діапазоні спектра широко застосовуються в різних сферах людської діяльності. Тепловізійні пристрої використовують термовипромінювання, які роблять можливим виділення предметів і людей в повній темряві та в різних складних несприятливих для візуального спостереження погодних умовах. Типовим застосуванням такої технології є системи безпеки і спостереження, а також галузь автономної навігації літальних апаратів і наземних транспортних засобів.

Метою роботи є огляд існуючих критеріїв для точної оцінки істинних характеристик ОЕС з урахуванням несприятливих факторів реальних умов.

II. ОСНОВИ ВІЗУАЛІЗАЦІЇ ІНФРАЧЕРВОНИХ ЗОБРАЖЕНЬ

Як вже було зазначено, тепловізійні системи дозволяють виявляти об'єкти в повній темряві. Це досягається завдяки можливості нагрітих об'єктів випромінювати або відбивати електромагнітні хвилі, в інфрачервоному (ІЧ) діапазоні. Інфрачервоне випромінювання має більшу довжину хвилі, ніж видима частина спектру, тому для людини таке випромінювання є непомітним [1].

Інфрачервоний спектр, в свою чергу, поділяється на кілька діапазонів [2]:

- Near-Infrared (NIR) – світло ближнього ІЧ діапазону спектра, від 0,7 до 1 мкм.
- Short-Wave Infrared (SWIR) – світло короткохвильового ІЧ діапазону спектра, від 1 до 3 мкм.

- Middle-Wave Infrared (MWIR) – світло середнього ІЧ діапазону спектра, від 3 до 5 мкм.
- Long-Wave Infrared (LWIR) – довгохвильовий діапазон ІЧ спектра, від 8 до 14 мкм.
- Very Long-Wave infrared (VLWIR) – понаддовгохвильовий діапазон ІЧ спектра, понад 14 мкм.

Тепловізори здатні приймати випромінювання об'єкту в ІЧ діапазонах спектру, які мають вікно прозорості атмосфери, тобто MWIR та LWIR [3], як показано на Рис. 1 [4]. У сучасних системах формування ІЧ зображень використовується довгохвильовий діапазон для збереження умов освітлення.

Перед тим як ІЧ зображення дійде до спостерігача, воно повинно пройти надскладну структуру, яка включає оптичні лінзи, матрицю, що реєструє інфрачервоне світло, блоки обробки зображення та оптимізації і виведення на дисплей.

Структурна схема процесу поетапного формування ІЧ зображень зображена на Рис. 2 [3].

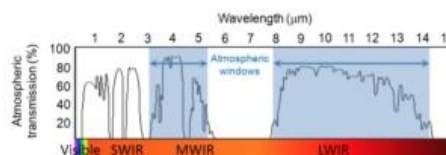


Рис. 1 ІЧ діапазони електромагнітного спектра, які є прозорими в повітрі





Рис. 2 Ланцюг формування ІЧ зображення в тепловізійних системах

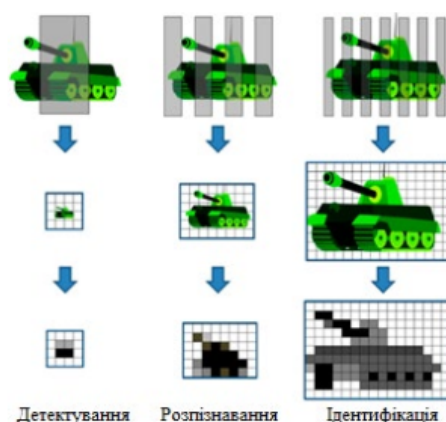


Рис. 3 Ілюстрація застосування методу на реальній цілі і передбачуваному зображенні

Таблиця 1 Рівні візуального сприйняття

Рівні візуального сприйняття	Роздільна здатність	
	Пари ліній на розмір цілі	Пікселі на розмір цілі
Виявлення	$1 \pm 0,25$	2
Орієнтація	$1,4 \pm 0,35$	-
Розпізнавання	$4 \pm 0,8$	7
Ідентифікація	$4 \pm 0,8$	13

III. ОСНОВНИЙ КРИТЕРІЙ ВИЯВЛЕННЯ ОБ'ЄКТІВ

На сьогоднішній день існує безліч методів виявлення об'єктів на ІЧ зображенні з використанням ОЕС, але першовідкривачем в цьому питанні став Джон Джонсон в 1958 році. Він охарактеризував ймовірність виявлення об'єкта на основі ефективної роздільної здатності зображуваного предмета і виявив, що в міру збільшення роздільної здатності цілі, зростає й ймовірність того, що спостерігач успішно її виявить [5].

В основі методу виявлення лежить певна кількість штрихових пар ліній, які заповнюють спостережуваний об'єкт в залежності від його роздільної здатності (Рис. 3, [3]). Існує чотири основних рівня візуального сприйняття, які визначив Джонсон: 1) виявлення (нааяність об'єкта), 2) визначення орієнтації (вертикальна, горизонтальна), 3) розпізнавання (визначення класу об'єкту), 4) ідентифікація (розрізнення об'єктів

в класі). Кожен рівень визначається своєю кількістю штрихованих пар ліній. Відповідно, якщо на об'єкт укладається більша кількість подібних пар, то такий об'єкт є більш інформативним. У Таблиці 1 наведено кількість пар ліній і пікселів на розмір цілі, які відповідають кожному рівню сприйняття.

Критерій Джонсона не враховує безліч факторів, які впливають на розпізнавання об'єктів, тобто він застосовується лише в ідеальних умовах, але даний концепт ліг в основу подальших удосконалених моделей і методів.

IV. ЗАГАЛЬНИЙ АНАЛІЗ ІСНУЮЧИХ ПРОБЛЕМ

Кожна ланка ланцюга формування зображення будь-яким чином впливає [6]–[8] на сприйняття об'єктів в тепловізійних пристроях. Перед тим як випромінювання від об'єкта досягне оптики тепловізійного сенсора, воно повинно пройти через безліч перешкод, які включають в себе атмосферні фактори такі як: опади і стан повітря (вологість, температура), сонячна радіація тощо. При цьому, ймовірність виявлення об'єкта при дальності в 10 км може падати на 30% [9], що показує, наскільки сильно несприятливі погодні умови впливають на розпізнавання і підкреслюють важливість їх врахування.

Крім погодних умов на ймовірність виявлення впливають візуальні перешкоди у вигляді сторонніх об'єктів, які не потребують розпізнавання. Це може бути нерівномірний фон зі змінною яскравістю або фон, що складається з геометрично схожих предметів. Джон Маз в своїй роботі [10] вивів кореляцію між зашумленим фоном в ІЧ спектрі і складністю виявлення цілі. При нерівномірному фоні кількість пар ліній на розмір об'єкту в ІЧ діапазоні може збільшуватися з 1,3 до 2,5 і відповідно для його розпізнавання знадобиться більша роздільна здатність.

При проектуванні ОЕС також варто враховувати параметр відношення сигнал/шум (SNR). В літературі [9], [11], [12] цей фактор розглядають разом з розміткою зображення, так як він аналогічним чином впливає на його розпізнавання. На сьогоднішній день навіть сучасні ОЕС схильні до втрати різкості зображення. Цей параметр визначається за допомогою функцією передачі модуляції.

Фактором впливу при виявленні цілі на ІЧ зображенні важливу роль відіграє сама цілі. Необхідно враховувати її геометричні розміри, зокрема відношення висоти об'єкта до його ширини, а також орієнтацію в просторі. Проведені дослідження, в яких проаналізовано вплив орієнтації танка на його виявлення, вказали, що ймовірність виявлення цілі збоку вище на 30%, ніж спереду [13], а ідентифікація літальних апаратів, в залежності від кута спостереження, може варіюватися до 60% [14].

Окрім геометричних характеристик об'єкту і його орієнтації, значний ступінь в розпізнаванні має спроможність визначення відмінних ознак, особливо при захопленні цілей. Відмінні ознаки в даному випадку варто приймати за унікальні властивості об'єкту і не враховувати загальні ознаки класу. Чим менше ідентифікаційних знаків у цілі, тим складніше її виявити.



Наведені вище чинники є основними серед тих, що впливають на оцінку виявлення об'єктів в реальних умовах, тому їх врахування є обов'язковим для створення якісної ОЕС. Щоб врахувати зазначені фактори для оцінювання якості оптичних систем були запропоновані моделі, розглянуті нижче.

V. МОДЕЛІ ДЛЯ ОЦІНЮВАННЯ ЯКОСТІ ОПТИКО-ЕЛЕКТРОННИХ СИСТЕМ

Для врахування вищезазначених факторів, критерій Джонсона розширювався і вводили нові показники. Початкове рівняння (1), яке описує ймовірність виявлення цілі і на якому базуються більш точні сучасні моделі, було запропоновано в 1974 році Джоном Джонсоном і Уолтером Лоусом [9]. Ця модель оцінювання має вигляд:

$$P(t) = P_{\infty} \left[1 - e^{-(t/\tau_{FOR})^E} \right], \quad (1)$$

$$P_{\infty} = \frac{(N/N_{50})^E}{1 + (N/N_{50})^E},$$

$$E = 2,7 + 0,7(N/(N_{50}D)),$$

де $P(t)$ – ймовірність виявлення у визначений момент часу t ; P_{∞} – ймовірність за весь час; E – масштабоване значення, знайдене в результаті експериментів; N – число періодів штрихових пар ліній, які заповнюють спостережуваний об'єкт; N_{50} – число періодів для ймовірності виявлення в 50%; t – момент часу в секундах; τ_{FOR} – середній час пошуку цілі в секундах.

Модель Target Task Performance (TTP) є досконалішою і враховує більшість зовнішніх факторів – характеристики цілі, її орієнтацію в просторі, а також параметр SNR і розмиття зображення з нерівномірним фоном. Такий підхід (2) забезпечує більш точні дані для розрахунку параметрів ОЕС [15]:

$$P = \frac{\left(\frac{N_{resolved}}{V_{50}} \right)^E}{1 + \left(\frac{N_{resolved}}{V_{50}} \right)^E} \quad (2)$$

$$N_{resolved} = \frac{\sqrt{A_{TGT} TTP}}{Range}$$

$$TTP = \int_{\xi_{low}}^{\xi_{cut}} \left[\frac{C_{TGT}}{CTF_{sys}(\xi)} \right]^{1/2} d\xi$$

$$C_{TGT} = \frac{\sqrt{\Delta\mu_{tgt}^2 + \sigma_{tgt}^2}}{2\mu_{scene}}$$

$$E = 1,51 + 0,24 \frac{N_{resolved}}{V_{50}},$$

де P – ймовірність виявлення цілі; $N_{resolved}$ – число періодів штрихових пар ліній, які заповнюють спостережуваний об'єкт; TTP – значення необхідне для

визначення якості зображення і прогнозування дальності виявлення; C_{TGT} – значення контрасту цілі на фоні; E – масштабоване значення критерію TTP; V_{50} – число періодів для ймовірності виявлення в 50%; A_{TGT} – квадратний корінь цільової області у метрах; $Range$ – дальність до цілі в кілометрах; ξ_{cut} – найвища роздільна здатність, яка може бути досягнута при заданому контрасті; ξ_{low} – найнижча роздільна здатність, яка може бути досягнута при заданому контрасті (звичай 0); $CTF_{sys}(\xi)$ – контрастна порогова функція системи; μ_{tgt} – середня задана температура цілі в °K; σ_{tgt} – стандартне відхилення заданої температури в °K; μ_{scene} – температура фону в °K.

Для розрахунку впливу погодних умов використовують програмні пакети щоб змоделювати атмосферні чинники за допомогою комп'ютерної симуляції. Бази даних моделей LOWTRAN і MODTRAN продовжують наповнюватись свіжою інформацією протягом кількох десятиків років і зараз вони доступні у вигляді модулів і пакетів для Python та MATLAB® [16]–[19].

ВИСНОВКИ

В роботі проаналізовано чинники, які можуть впливати на відстань роботи та здатність виявляти об'єкти в тепловізійних ОЕС. Встановлено, що найбільший негативний внесок мають фактори погоди і атмосферні чинники, нерівномірний фон в сцені, розмиття або розфокусування зображення, а також характеристики самої цілі – її розміри і орієнтація в просторі. Розглянуто моделі та програмні пакети, які дозволяють визначити реальні характеристики розроблюваного пристрою і виявити слабку ланку приладу.

ПЕРЕЛІК ПОСИЛАНЬ

- [1] C. Solomon and T. Breckon, Fundamentals of Digital Image Processing, Chichester, UK: John Wiley & Sons, Ltd, 2010, ISBN: 9780470689776.
- [2] T. Bourlai and B. Cukic, "Multi-spectral face recognition: Identification of people in difficult environments," in 2012 IEEE International Conference on Intelligence and Security Informatics, 2012, pp. 196–201, DOI: [10.1109/ISI.2012.6284307](https://doi.org/10.1109/ISI.2012.6284307).
- [3] D. Perić, B. Livada, M. Perić, and S. Vujić, "Thermal Imager Range: Predictions, Expectations, and Reality," Sensors, vol. 19, no. 15, p. 3313, Jul. 2019, DOI: [10.3390/s19153313](https://doi.org/10.3390/s19153313).
- [4] "Infrared – The Light We Cannot See," IRG Educational Series, 2016. [Online]. Available: <http://www.irradiance-glass.com/2016/08/09/irg-educational-series-infrared-the-light-we-cannot-see/>. [Accessed: 09-Aug-IBC].
- [5] J. Johnson, "Analysis of image forming systems," in Proceedings of the Image Intensifier Symposium, 1958, URL: <https://home.cis.rit.edu/~cnspci/references/johnson1958.pdf>.
- [6] M. van Iersel et al., "The influence of meteorological parameters on dynamic infrared signatures," 2016, p. 997907, DOI: [10.1117/12.2239474](https://doi.org/10.1117/12.2239474).
- [7] N. S. Kopeika, A. N. Seidman, I. Dinstein, C. Tamasha, R. Amir, and Y. Biton, "How Weather Affects Seeing Through The Atmosphere," Opt. Eng., vol. 25, no. 3, p. 253505, Mar. 1986, DOI: [10.1117/12.7973850](https://doi.org/10.1117/12.7973850).



- [8] B. Wang, L. Dong, M. Zhao, H. Wu, Y. Ji, and W. Xu, "An infrared maritime target detection algorithm applicable to heavy sea fog," *Infrared Phys. Technol.*, vol. 71, pp. 56–62, Jul. 2015, DOI: [10.1016/j.infrared.2015.01.031](https://doi.org/10.1016/j.infrared.2015.01.031).
- [9] T. A. Sjaardema, C. S. Smith, and G. C. Birch, "History and Evolution of the Johnson Criteria," Albuquerque, NM, and Livermore, CA (United States), Jul. 2015, URL: <http://www.osti.gov/servlets/purl/1222446/>.
- [10] J. P. Mazz, "Analysis of observer variability in the assessment of FLIR performance," in *Infrared Imaging Systems: Design, Analysis, Modeling, and Testing VII*, 1996, pp. 2–11, DOI: [10.1117/12.241975](https://doi.org/10.1117/12.241975).
- [11] K. Krapels, R. G. Driggers, D. Deaver, S. K. Moyer, and J. Palmer, "Midwave infrared and visible sensor performance modeling: small craft identification discrimination criteria for maritime security," *Appl. Opt.*, vol. 46, no. 30, p. 7345, Oct. 2007, DOI: [10.1364/AO.46.007345](https://doi.org/10.1364/AO.46.007345).
- [12] K. Krapels et al., "Small craft ID criteria (N50/V50) for short wave infrared sensors in maritime security," *Infrared Imaging Syst. Des. Anal. Model. Test. XIX*, vol. 6941, p. 694108, 2008, DOI: [10.1117/12.778062](https://doi.org/10.1117/12.778062).
- [13] S. R. Rotman, "Modeling human search and target acquisition performance: 3. target detection in the presence of obscursants," *Opt. Eng.*, vol. 30, no. 6, p. 824, Jun. 1991, DOI: [10.1117/12.55852](https://doi.org/10.1117/12.55852).
- [14] J. C. Leachtenauer and R. G. Driggers, *Surveillance and Reconnaissance Imaging Systems-Modeling and Performance Prediction*. Boston, London: Artech House, 2001, ISBN: 1-58053-132-6.
- [15] R. H. Vollmerhausen and E. Jacobs, "The Targeting Task Performance (TTP) Metric A New Model for Predicting Target Acquisition Performance," Apr. 2004, URL: <http://www.dtic.mil/docs/citations/ADA422493>.
- [16] "MODTRAN Infrared Light in the Atmosphere." [Online]. Available: <http://climatemodels.uchicago.edu/modtran/modtran.doc.html>.
- [17] D. Griffith, "Matlab Class Wrapper for MODTRAN 4." [Online]. Available: <https://www.mathworks.com/matlabcentral/fileexchange/26927-matlab-class-wrapper-for-modtran-4>.
- [18] M. Noah, "LOWTRAN7 Sun and Moon Models." [Online]. Available: <https://www.mathworks.com/matlabcentral/fileexchange/71203-lowtran7-sun-and-moon-models>.
- [19] M. Hirsch, "LOWTRAN: Python module for atmospheric absorption modeling," 2016, DOI: [10.5281/zenodo.213475](https://doi.org/10.5281/zenodo.213475).

UDC 621.384.3

Analysis of Factors Affecting the Efficiency of Electro-Optical Systems

A. S. Lysenko

Department of Design of Electronic Digital Equipment keoa.kpi.uaNational Technical University of Ukraine «Igor Sikorsky Kyiv Polytechnic Institute» kpi.ua
Kyiv, Ukraine

Abstract—According to present knowledge thermal imaging electro-optical surveillance systems with object detection are widely used in various fields of human activity. Thermal imagers use infrared radiation emitted by objects and people, so it makes possible to take a sight in complete darkness and miscellaneous weather conditions. Typical applications of this technology are security and surveillance systems, along with autonomous navigation industries in aircraft and ground vehicle. The aim of the work is to determine the necessary condition for an accurate parameter estimation of the electro-optical systems allowing for adverse externalities.

First of all, the various infrared spectrum ranges will be considered and corresponding wavelengths will be pointed for each of the following. Thermal imaging cameras are capable of receiving emission from an object in the infrared spectral ranges, which have an atmospheric transparency window, i.e. MWIR and LWIR.

Then the structural diagram of the phased generation process of infrared images in thermal imaging systems will be discussed. It appears that, the image formation chain includes such components as irradiation source, optics, imaging sensor, read-out electronics, processing electronics, compression and transmission of data, decoding, search & track engine, display and finally human observer.

The next chapter will describe the probability method of detecting objects based on the effective resolution of the depicted object. There will be resulted the table of visual perception levels, namely orientation, recognition, identification. And relative levels of perception will be provided the corresponding resolution of the image.

The next section will analyze the factors affecting the objects identification. Before the radiation from the object reaches the thermal imaging sensor optics, it must go through many obstacles, which include atmospheric factors such as precipitation and air condition (humidity, temperature), solar radiation, etc. Various equations and models will be carried out to take into account the above factors. For example, Johnson's criterion, and a more advanced Target Task Performance recognition model.

This paper considers the main criteria for detecting objects using electro-optical systems in the infrared range. The robust analysis of literature sources will be conduct and describe the basics of visualization and the infrared imaging in thermal systems. Then Johnson criteria, which is considering for calculating the efficiency of electro-optical systems, will be evaluated for validity using in real-world environment. Also, it will be pointed out, that the greatest negative contribution on visualization is made by weather and atmospheric conditions, clutter effects and density, motion-blur and defocusing, along with target properties – aspect ratio and viewing angle. Finally, it will be suggested prediction models and software packages that will help determine the real characteristics of the elaborate device and identify the weak link of the system.

Keywords — surveillance systems; thermal imaging; infrared image; object detection.



Copyright (c) 2020 Лисенко А. С.

Додаток Б

Функція конвертації зображень coco_2_YOLO.py:

```
from __future__ import print_function
import argparse
import glob
import os
import sys
import json

if __name__ == '__main__':
    parser = argparse.ArgumentParser()
    parser.add_argument(
        "path", help=DIR to .json)
    parser.add_argument(
        "output_path", help=DIR to .txt)
    args = parser.parse_args()
    json_files = sorted(glob.glob(os.path.join(args.path,
        '*.json')))
    if args.debug:
        total_count = 0
        categorys = {0: 0, 1: 0, 2: 0}
        bike_images = set()
    for json_file in json_files:
        with open(json_file) as f:
            data = json.load(f)
            image = data['image']
            annotations = data['annotation']
            file_name = image['file_name']
            width = float(image['width'])
            height = float(image['height'])
            converted_results = []
            for ann in annotations:
                cat_id = int(ann['category_id'])
```

```

if cat_id <= 3:
    left, right, bbox_width, bbox_height = map(
        float, ann['bbox'])

    # Classes start from
    cat_id -= 1
    if args.debug:
        categorys[cat_id] += 1
        total_count += 1
        if cat_id == 1:
            bike_images.add(file_name)
    x_center, y_center = (
        left + bbox_width / 2, right +
bbox_height / 2)

    # YOLO expects rel val from width x height
of img
    x_rel, y_rel = (x_center / width, y_center /
height)
    w_rel, h_rel = (bbox_width / width,
bbox_height / height)
    converted_results.append(
        (cat_id, x_rel, y_rel, w_rel, h_rel))

    image_name = img[i]['file_name']
    image_name = image_name[14:-5]
    print(image_name)
    file = open(args.output_path + str(image_name) +
'.txt', 'w+')
    file.write('\n'.join('%d %.6f %.6f %.6f %.6f' %
res for res in converted_results))
    #file.close()

if args.debug:
    print({cat: categorys[cat] for cat in categorys})

```

```

        print(total_count)
        with open('bikes.txt', 'a+') as f:
            f.write('\n'.join("data/thermal/%s.jpeg"%b_img      for
b_img in bike_images))
            f.write('\n')

```

Jupyter файл YOLOtiny_FLIR_MY.ipynb для тренування в Colab:

```

# -*- coding: utf-8 -*-
"""YOLOtiny_FLIR_MY.ipynb

Automatically generated by Colaboratory.

# Step 1: Cloning and Building Darknet
"""

# check the GPU
!nvidia-smi --query-gpu=gpu_name,driver_version,memory.total --
format=csv

"""Clone darknet from AlexeyAB's repository"""

# clone darknet repo
!git clone https://github.com/AlexeyAB/darknet

"""Adjust the Makefile to enable OPENCV and GPU for darknet."""

# Commented out IPython magic to ensure Python compatibility.
# change makefile to have GPU and OPENCV enabled
# %cd darknet
!sed -i 's/OPENCV=0/OPENCV=1/' Makefile
!sed -i 's/GPU=0/GPU=1/' Makefile
!sed -i 's/CUDNN=0/CUDNN=1/' Makefile
!sed -i 's/CUDNN_HALF=0/CUDNN_HALF=1/' Makefile

```

```

# verify CUDA
!/usr/local/cuda/bin/nvcc --version

# make darknet
!make

"""# STEP 2: Google Drive
Images will be uploaded from Google Drive

We will create a symbolic link between '/content/gdrive/My\
Drive/' and '/mydrive.
"""

# Commented out IPython magic to ensure Python compatibility.
# %cd ..
from google.colab import drive
drive.mount('/content/gdrive')

# this creates a symbolic link so that now the path
/content/gdrive/My\ Drive/ is equal to /mydrive
!ln -s /content/gdrive/My\ Drive/ /mydrive
!ls /mydrive

# Commented out IPython magic to ensure Python compatibility.
# cd back into the darknet folder
# %cd darknet

"""# Step 3: Moving Your Custom Datasets Into Your Cloud VM

Training dataset folder with images and text files called
'**obj**' and it's uploaded to my GDrive.

The same with my validation dataset named '**test**', that also
uploaded to GDrive.

```

Copy in the zips and unzip them in my cloud VM.

"""

```
# this is where my datasets are stored within my Google Drive
```

```
!ls /mydrive/Diser/YOLOv4tiny
```

```
# copy over both datasets into the root directory of the Colab VM (commented out !remove commands)
```

```
!cp /mydrive/Diser/YOLOv4/obj.zip ../
```

```
!cp /mydrive/Diser/YOLOv4/test.zip ../
```

```
#!rm ../obj.zip
```

```
#!rm ../test.zip
```

```
# unzip the datasets and their contents so that they are now in /darknet/data/ folder
```

```
!unzip ../obj.zip -d data/
```

```
!unzip ../test.zip -d data/
```

```
#!rm -r data/obj
```

```
#!rm -r data/test
```

```
"""># Step 4: Configuring Files for Training
```

```
Configuring custom .cfg, obj.data, obj.names, train.txt and test.txt files.
```

```
## a) obj.names and obj.data
```

```
Created a new file within a text editor called **obj.names** where I will have all classes names.
```

"""

```
# upload the obj.names and obj.data files to cloud VM from Google Drive
```

```
!cp /mydrive/Diser/YOLOv4tiny/obj.names ./data
```

```
!cp /mydrive/Diser/YOLOv4tiny/obj.data ./data
```

```
"""># b) Generating train.txt and test.txt
```

train.txt and test.txt files which hold the relative paths to all our training images and valid images.

I have created scripts that easily generate these two files with proper paths to all images.

```
"""
```

```
# upload the generate_train.py and generate_test.py script to cloud VM from Google Drive
```

```
!cp /mydrive/Diser/YOLOv4tiny/generate_train.py ./
```

```
!cp /mydrive/Diser/YOLOv4tiny/generate_test.py ./
```

```
"""Run both scripts for generating the two txt files."""
```

```
!python generate_train.py
```

```
!python generate_test.py
```

```
"""Now we will create anchor boxes relative to our dataset"""
```

```
!./darknet detector calc_anchors data/obj.data -num_of_clusters 6 -width 512 -height 512
```

```
"""## c) .Cfg File
```

```
Copy over the yolov4.cfg to Google Drive by running the cell below. This will allow us to edit it in a text editor.
```

```
"""
```

```
# download cfg to google drive and change its name
```

```
!cp                                cfg/yolov4-tiny-custom.cfg /mydrive/Diser/YOLOv4tiny/yolov4tiny-obj.cfg
```

```
# to download to local machine (change its name to yolov4-obj.cfg once downloaded)
```

```
from google.colab import files
```

```
files.download('cfg/yolov4-custom.cfg')
```

```

# after editing, upload the custom .cfg back to cloud VM from
Google Drive

!cp /mydrive/Diser/YOLOv4tiny/yolov4tiny-obj.cfg ./cfg
#!rm ./cfg/yolov4tiny-obj.cfg

# verify that the newly generated train.txt and test.txt can be
seen in our darknet/data folder

!ls data/

"""# Step 5: Download pre-trained weights for the convolutional
layers.

This step downloads the weights for the convolutional layers of
the YOLOv4/Tiny network. (commented out YOLOv4 weights)

"""

#!wget
https://github.com/AlexeyAB/darknet/releases/download/darknet_yo
lo_v3_optimal/yolov4.conv.137

!wget
https://github.com/AlexeyAB/darknet/releases/download/darknet_yo
lo_v4_pre/yolov4-tiny.conv.29

"""# Step 6: Train Custom Object Detector

We are now ready to train custom YOLOv4 object detector. (-
dont_show flag stops chart from popping up since Colab Notebook
can't open images on the spot, -map flag overlays mean average
precision on chart to see how accuracy our model is)

"""

# train custom detector (uncomment %%capture below if there are
memory issues or Colab is crashing)

# %%capture

!./darknet detector train data/obj.data cfg/yolov4-my_obj.cfg
yolov4.conv.137 -dont_show -map

!./darknet detector train data/obj.data cfg/yolov4tiny-obj.cfg
yolov4-tiny.conv.29 -dont_show -map

"""# Step 7: Define Helper Functions

```

After training, we can observe a chart of how our model did throughout the training process. It shows a chart of average loss vs. iterations.

These three functions are helper functions that will allow to show the image in Colab Notebook after running detections, as well as upload and download images to and from Cloud VM.

```
"""
```

```
# Commented out IPython magic to ensure Python compatibility.
```

```
# define helper functions
```

```
def imShow(path):
```

```
    import cv2
```

```
    import matplotlib.pyplot as plt
```

```
#    %matplotlib inline
```

```
    image = cv2.imread(path)
```

```
    height, width = image.shape[:2]
```

```
    resized_image = cv2.resize(image, (3*width, 3*height),  
interpolation = cv2.INTER_CUBIC)
```

```
    fig = plt.gcf()
```

```
    fig.set_size_inches(18, 10)
```

```
    plt.axis("off")
```

```
    plt.imshow(cv2.cvtColor(resized_image, cv2.COLOR_BGR2RGB))
```

```
    plt.show()
```

```
# use this to upload files
```

```
def upload():
```

```
    from google.colab import files
```

```
    uploaded = files.upload()
```

```
    for name, data in uploaded.items():
```

```
        with open(name, 'wb') as f:
```

```
            f.write(data)
```



```

    print ('saved file', name)

# use this to download a file
def download(path):
    from google.colab import files
    files.download(path)

"""Here is what the chart.png could look like after an
**uninterrupted** training."""

# show chart.png of how custom object detector did with training
imshow('chart.png')

"""If for some reason we have got an error or Colab goes idle
during training, We can kick off training from our last saved
weights file stored in GDrive, so that we don't have to restart
from scratch. Every 100 iterations a weights file called
**yolov4-obj_last.weights** is saved to
**mydrive/yolov4/backup/** folder (where is my backup
folder)."""

# kick off training from where it last saved
!./darknet detector train data/obj.data cfg/yolov4tiny-obj.cfg
/mydrive/Diser/YOLOv4tiny/backup/yolov4tiny-obj_last.weights -
dont_show -map

"""# Step 8: Checking the Mean Average Precision (mAP) of Your
Model

We can find out the mAP of your model after run the following
command on any of the saved weights from the training to see the
mAP value for that specific weight's file.

"""

!./darknet detector map data/obj.data cfg/yolov4tiny-obj.cfg
/mydrive/Diser/YOLOv4tiny/backup/yolov4tiny-obj_4000.weights

"""# Step 9: Run Our Object Detector

We now have a custom object detector to make our own detections.

```

"""

```
# Commented out IPython magic to ensure Python compatibility.
# need to set our custom cfg to test mode (batch = 1)
# %cd cfg
!sed -i 's/batch=64/batch=1/' yolov4tiny-obj.cfg
!sed -i 's/subdivisions=16/subdivisions=1/' yolov4tiny-obj.cfg
# %cd ..

# run our custom detector with this command (an image uploaded
to my GDrive, thresh flag sets accuracy that detection must be
in order to show it)

!./darknet detector test data/obj.data cfg/yolov4tiny-obj.cfg
/mydrive/Diser/YOLOv4tiny/backup/yolov4tiny-obj_best.weights
/mydrive/Diser/YOLOv4/images/FLIR_10101.jpeg -thresh 0.5
imshow('predictions.jpg')
```